





Suren Manvelyan



By Suren Manvelyan, <http://www.surenmanvelyan.com/gallery/7116>

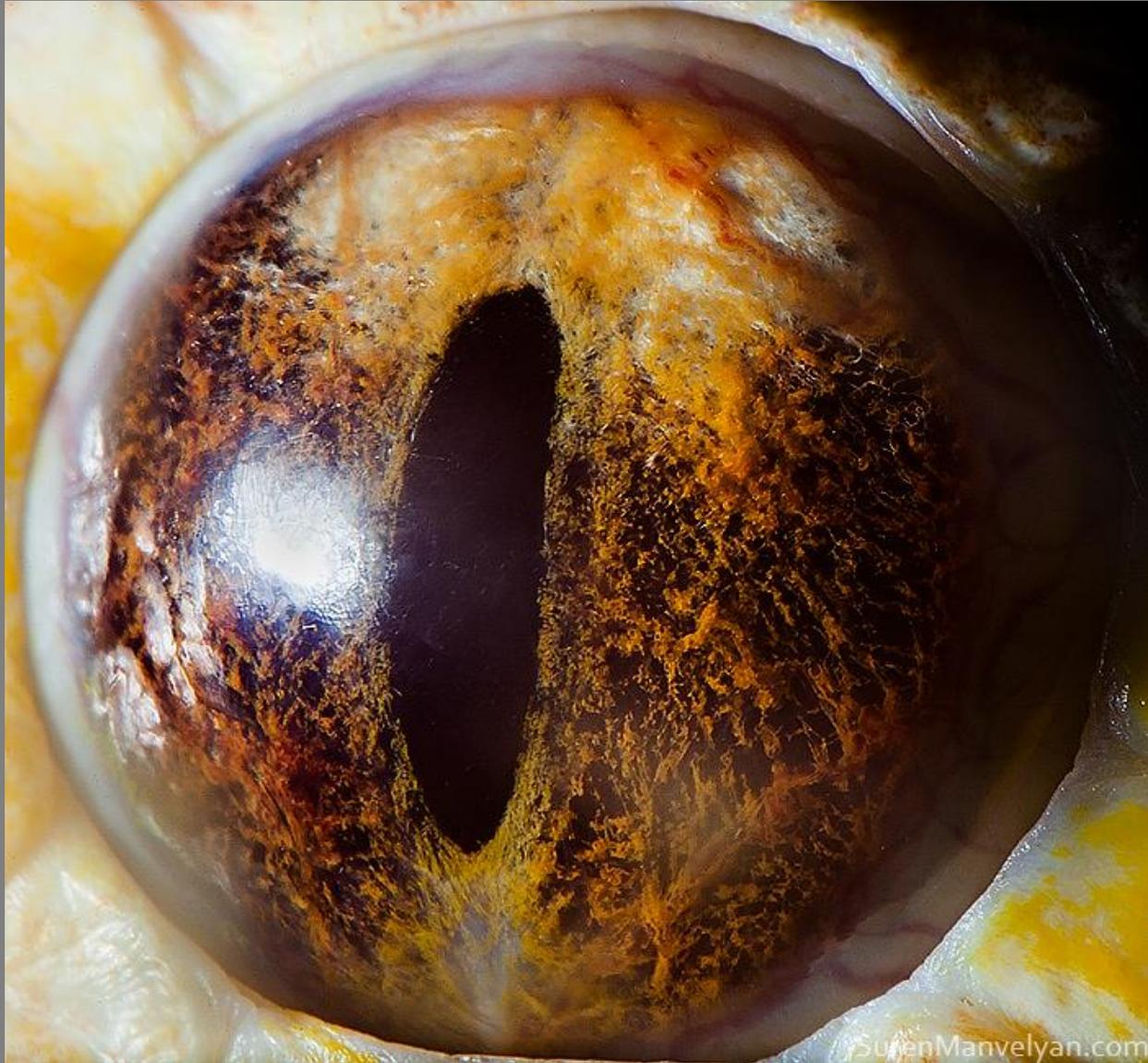


Suren Manvelyan



Suren Manvelyan

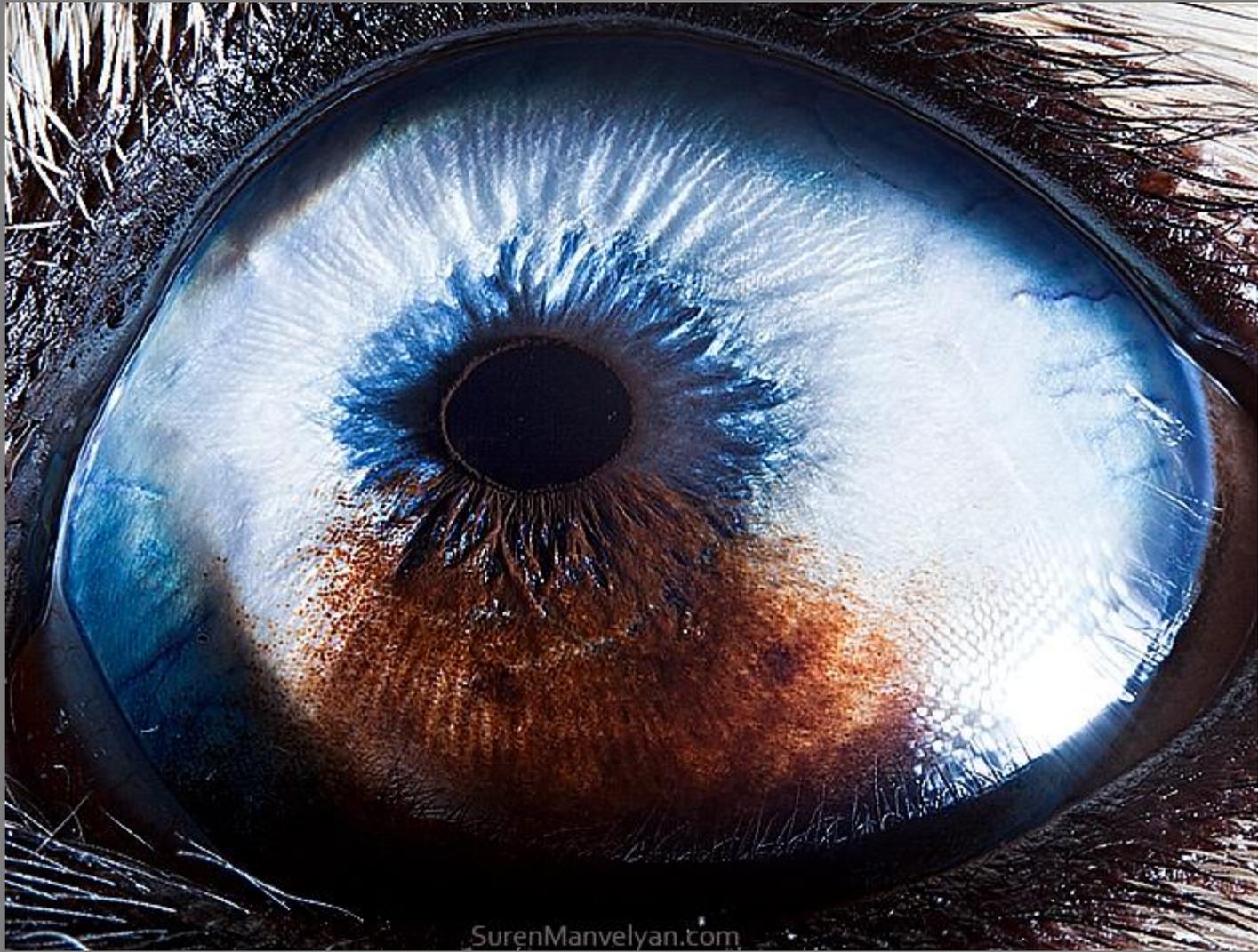




SurenManvelyan.com



Suren Manvelyan



SurenManvelyan.com





# Heterochromia iridum

From Wikipedia, the free encyclopedia

*Not to be confused with [Heterochromatin](#) or [Dichromatic \(disambiguation\)](#).*

In anatomy, **heterochromia** ([ancient Greek](#): ἕτερος, *héteros*, different + χρώμα, *chróma*, color<sup>[1]</sup>) is a difference in [coloration](#), usually of the [iris](#) but also of [hair](#) or [skin](#).

Heterochromia is a result of the relative excess or lack of [melanin](#) (a [pigment](#)). It may be [inherited](#), or caused by genetic [mosaicism](#), [chimerism](#), [disease](#), or [injury](#).<sup>[2]</sup>

Heterochromia of the [eye](#) (***heterochromia iridis*** or ***heterochromia iridum***) is of three kinds. In *complete heterochromia*, one iris is a different color from the other. In *sectoral heterochromia*, part of one iris is a different color from its remainder and finally in "central heterochromia" there are spikes of different colours radiating from the pupil.

## Heterochromia

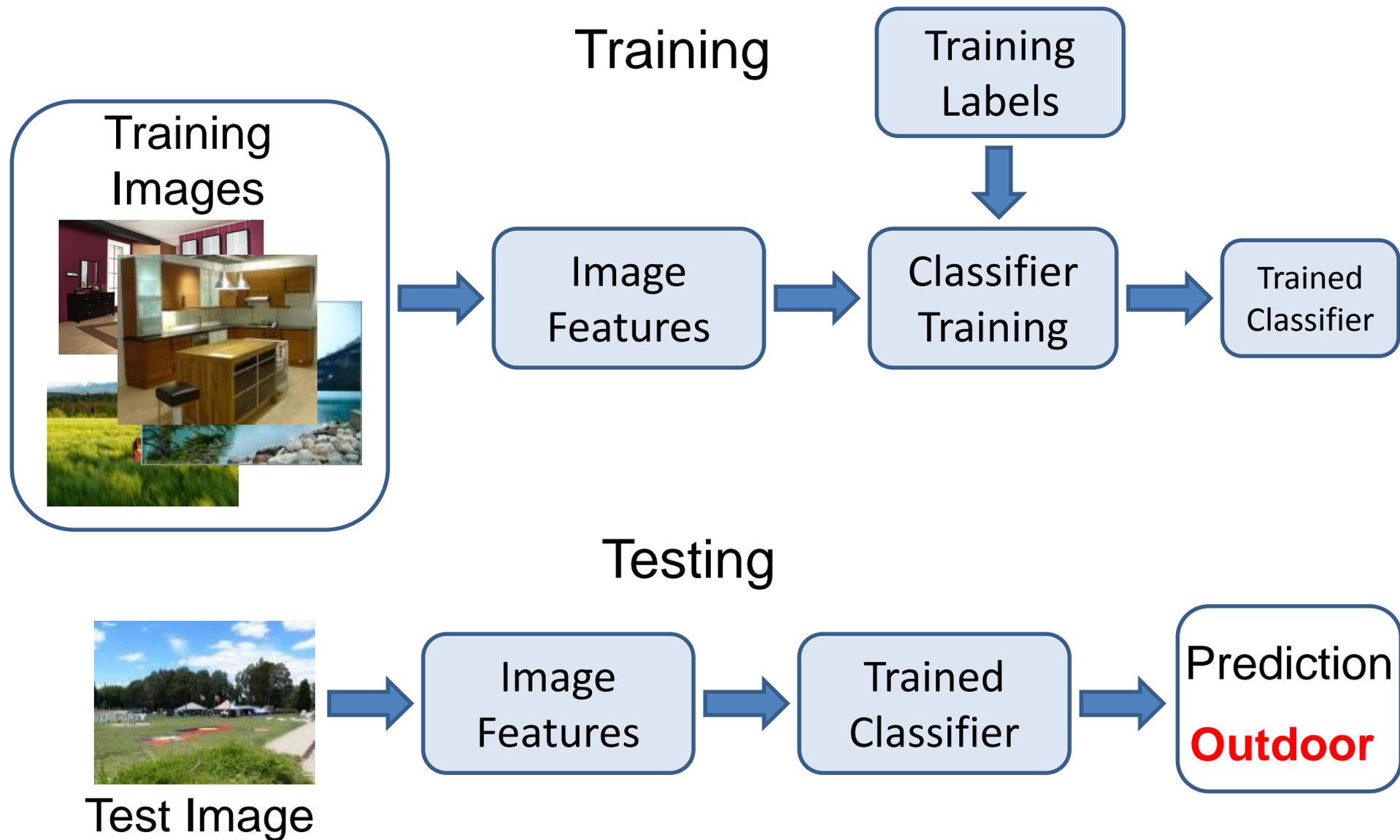


Complete heterochromia in human eyes: one brown and one green/hazel

### Classification and external resources

<b>Specialty</b>	<a href="#">ophthalmology</a>
<b>ICD-10</b>	<a href="#">Q13.2</a> , <a href="#">H20.8</a> , <a href="#">L67.1</a>
<b>ICD-9-CM</b>	<a href="#">364.53</a>
<b>OMIM</b>	<a href="#">142500</a>
<b>DiseasesDB</b>	<a href="#">31289</a>

# Image Categorization

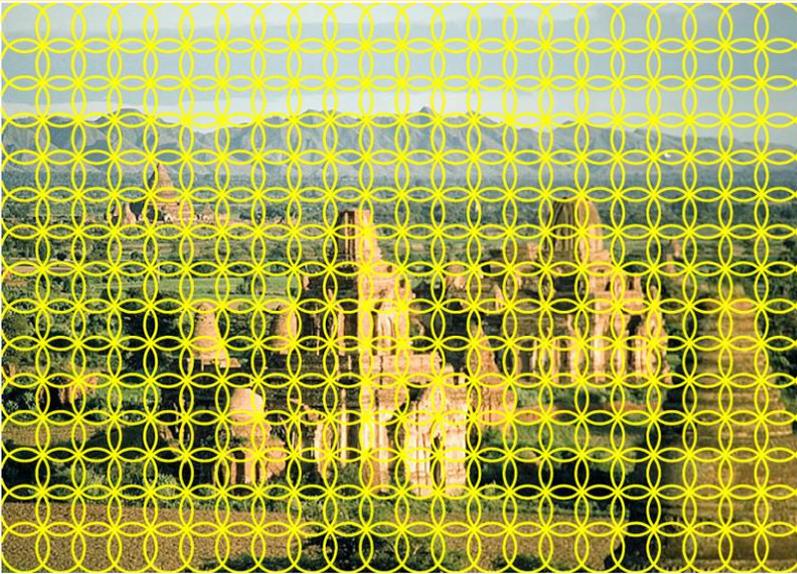


# History of ideas in recognition

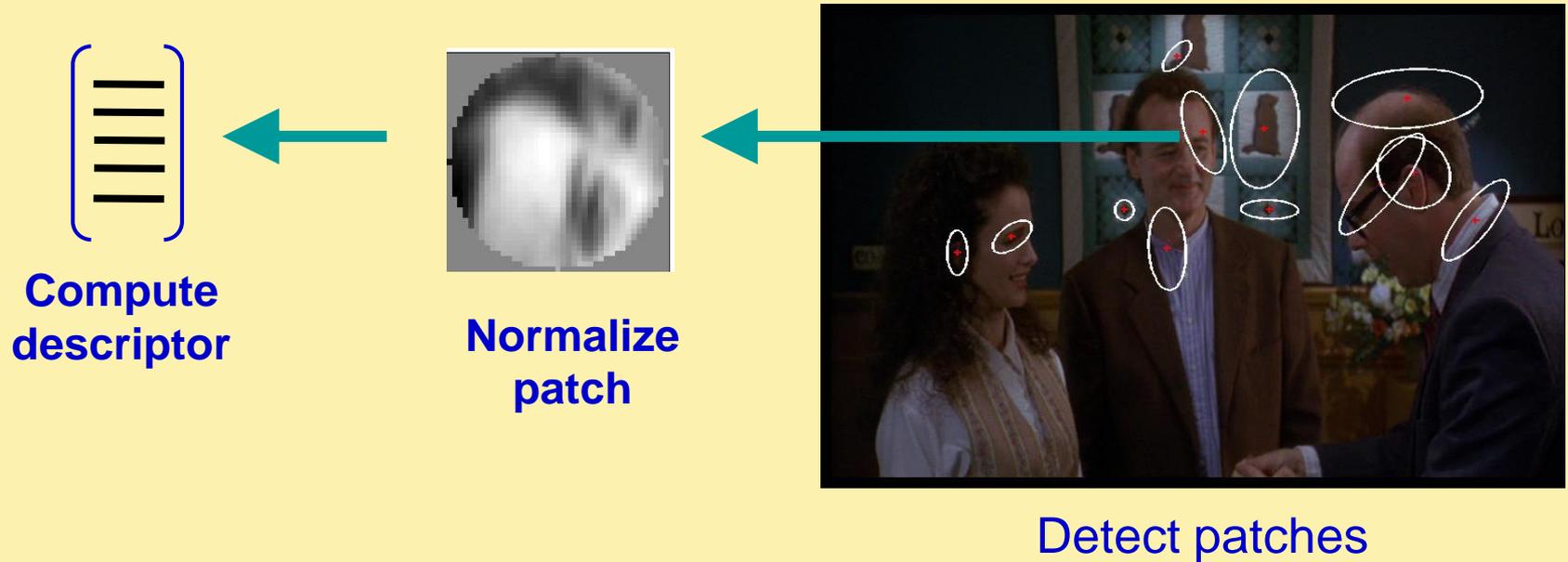
- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- Mid-1990s: sliding window approaches
- Late 1990s: local features
- Early 2000s: parts-and-shape models
- Mid-2000s: bags of features

# 1. Feature extraction

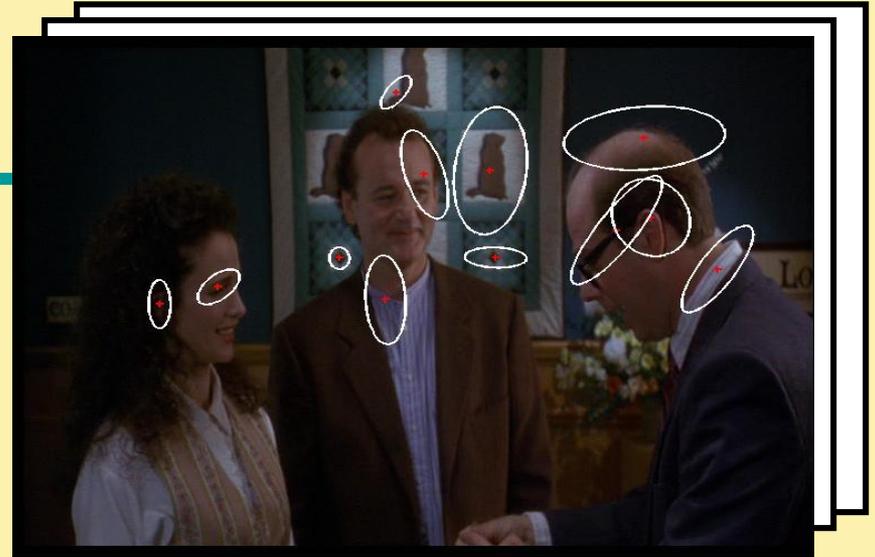
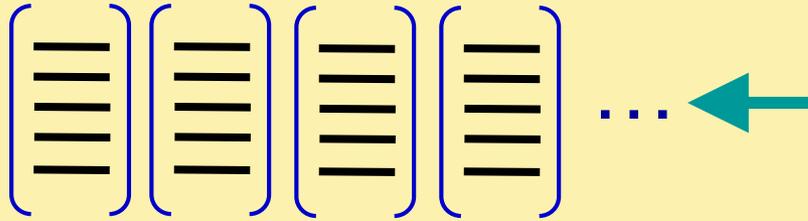
- Regular grid or interest regions



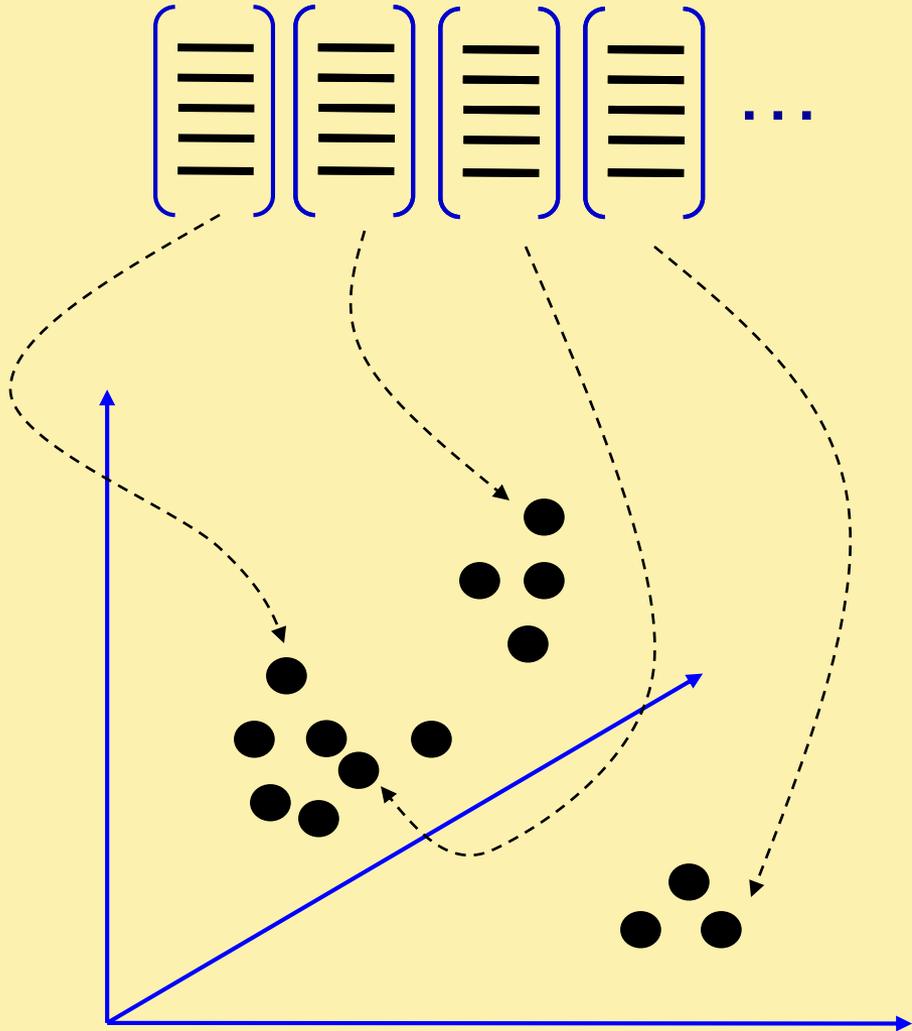
# 1. Feature extraction



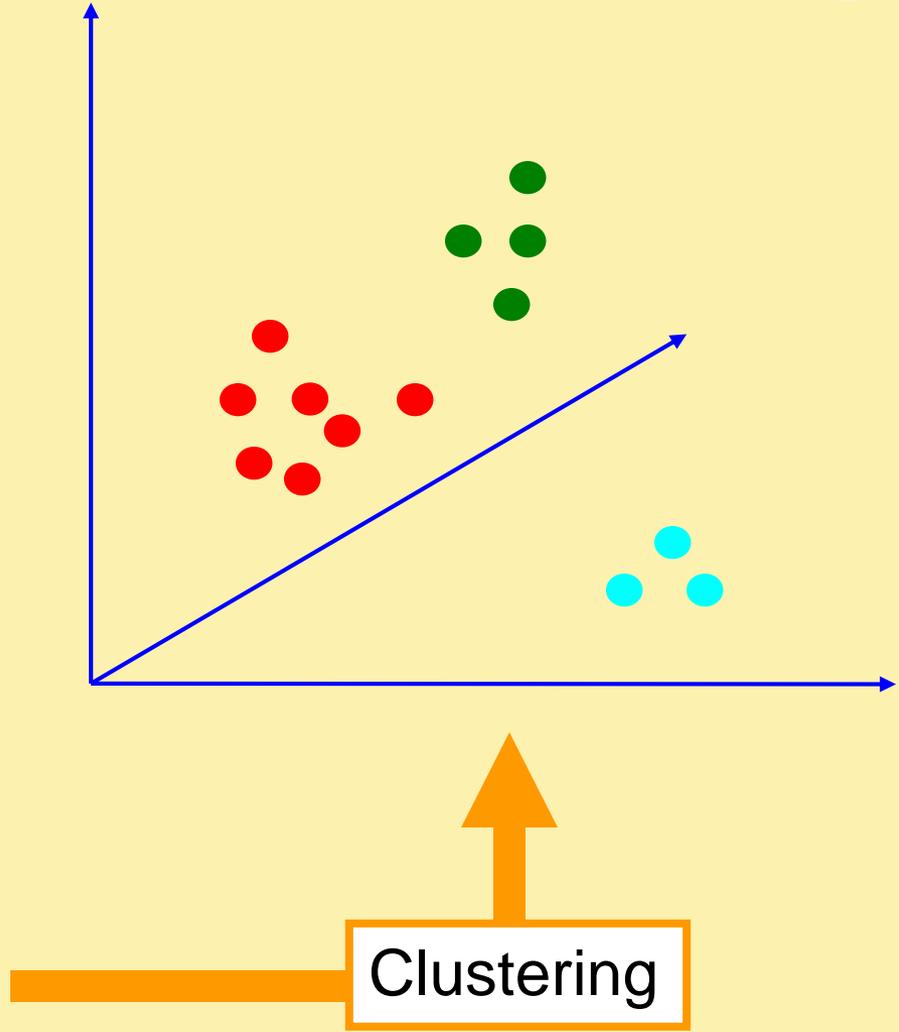
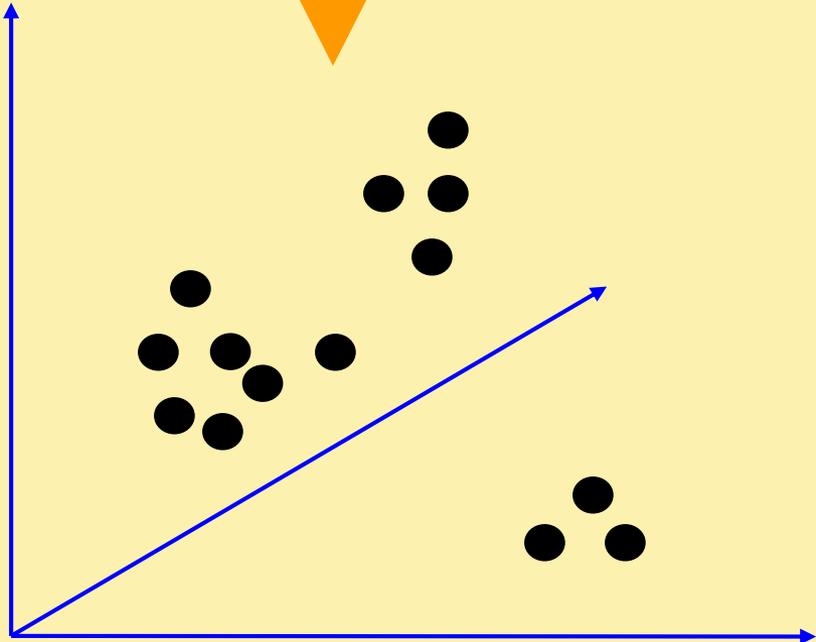
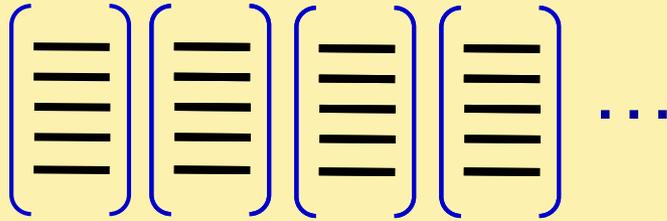
# 1. Feature extraction



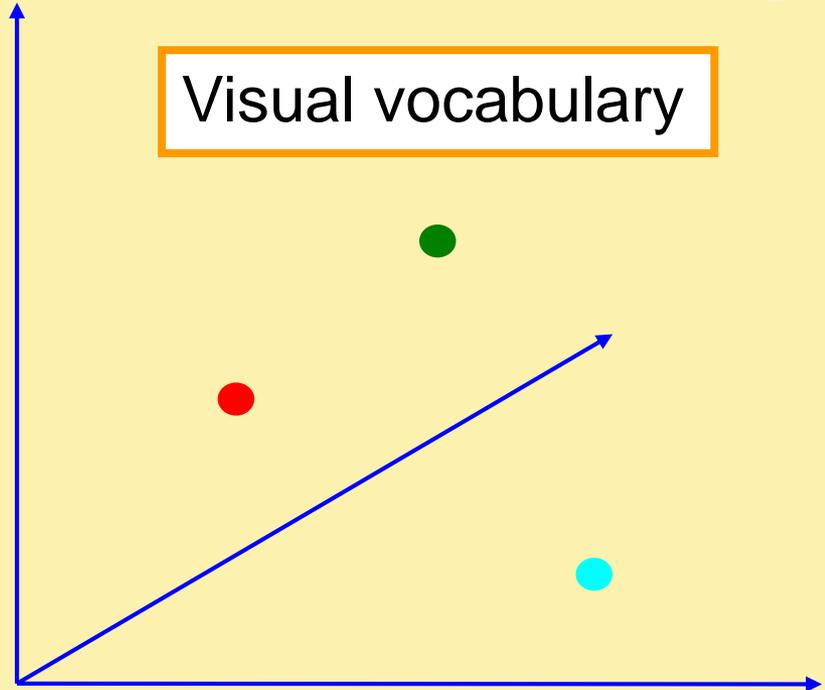
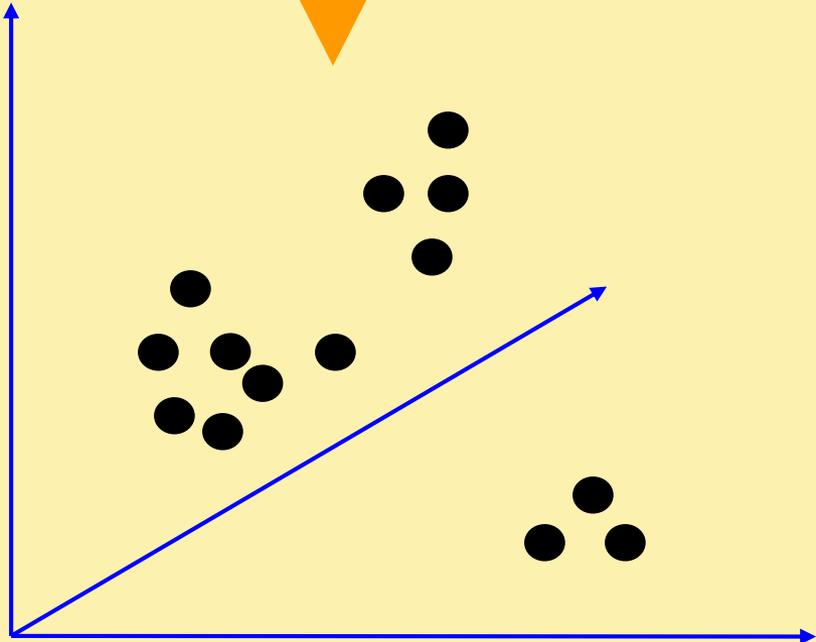
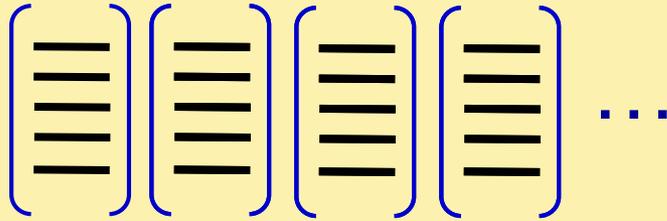
# 2. Learning the visual vocabulary



# 2. Learning the visual vocabulary



# 2. Learning the visual vocabulary

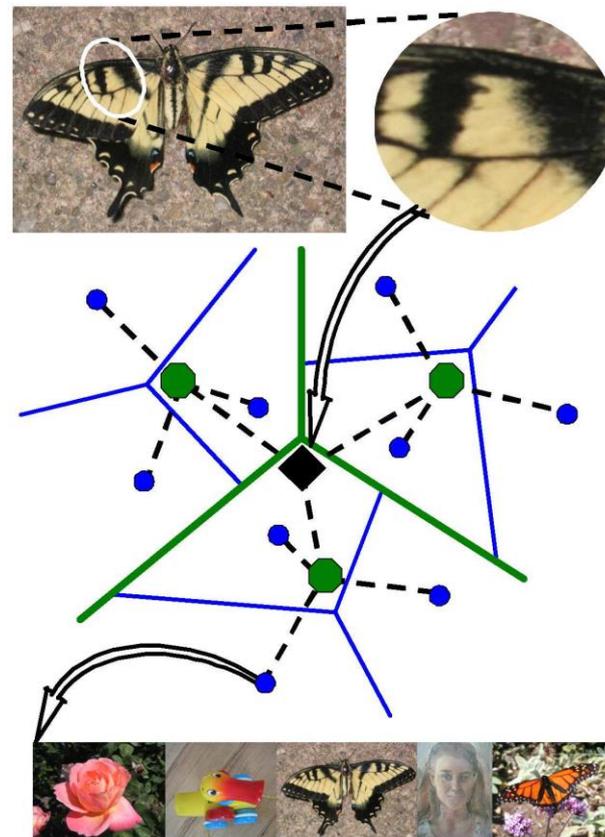




# Visual vocabularies: Issues

---

- How to choose vocabulary size?
  - Too small: visual words not representative of all patches
  - Too large: quantization artifacts, overfitting
- Computational efficiency
  - Vocabulary trees  
(Nister & Stewenius, 2006)



# History of ideas in recognition

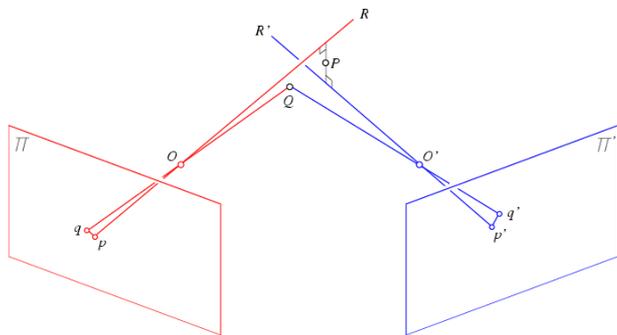
- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- Mid-1990s: sliding window approaches
- Late 1990s: local features
- Early 2000s: parts-and-shape models
- Mid-2000s: bags of features
- Present trends: combination of local and global methods, context, *deep learning*

# Large-scale Instance Retrieval

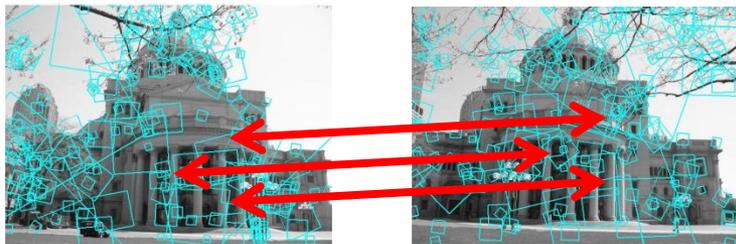
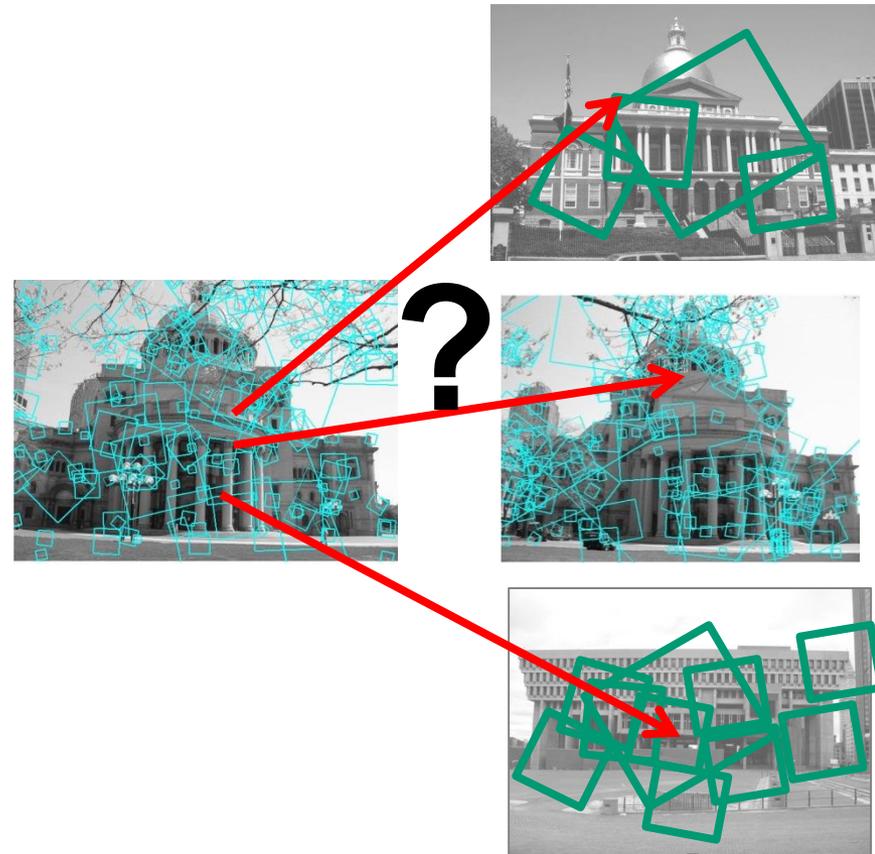
Computer Vision

James Hays

# Multi-view matching



vs



Matching two given views for depth

Search for a matching view for recognition

# Demo

# Video Google System

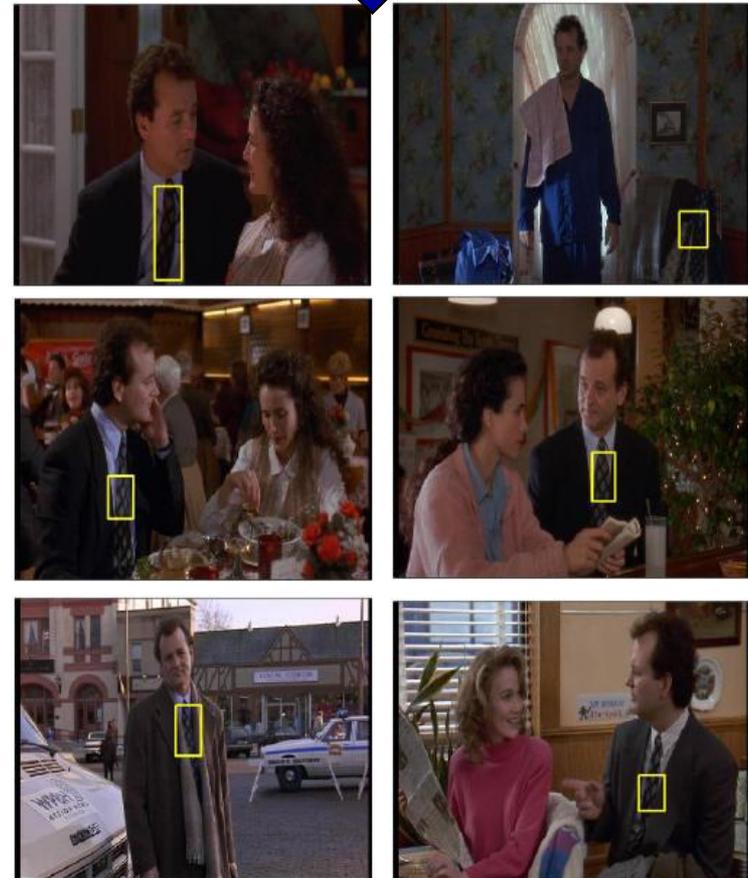
1. Collect all words within query region
2. Inverted file index to find relevant frames
3. Compare word counts
4. Spatial verification

Sivic & Zisserman, ICCV 2003

- Demo online at :  
<http://www.robots.ox.ac.uk/~vgg/research/vgoogle/index.html>

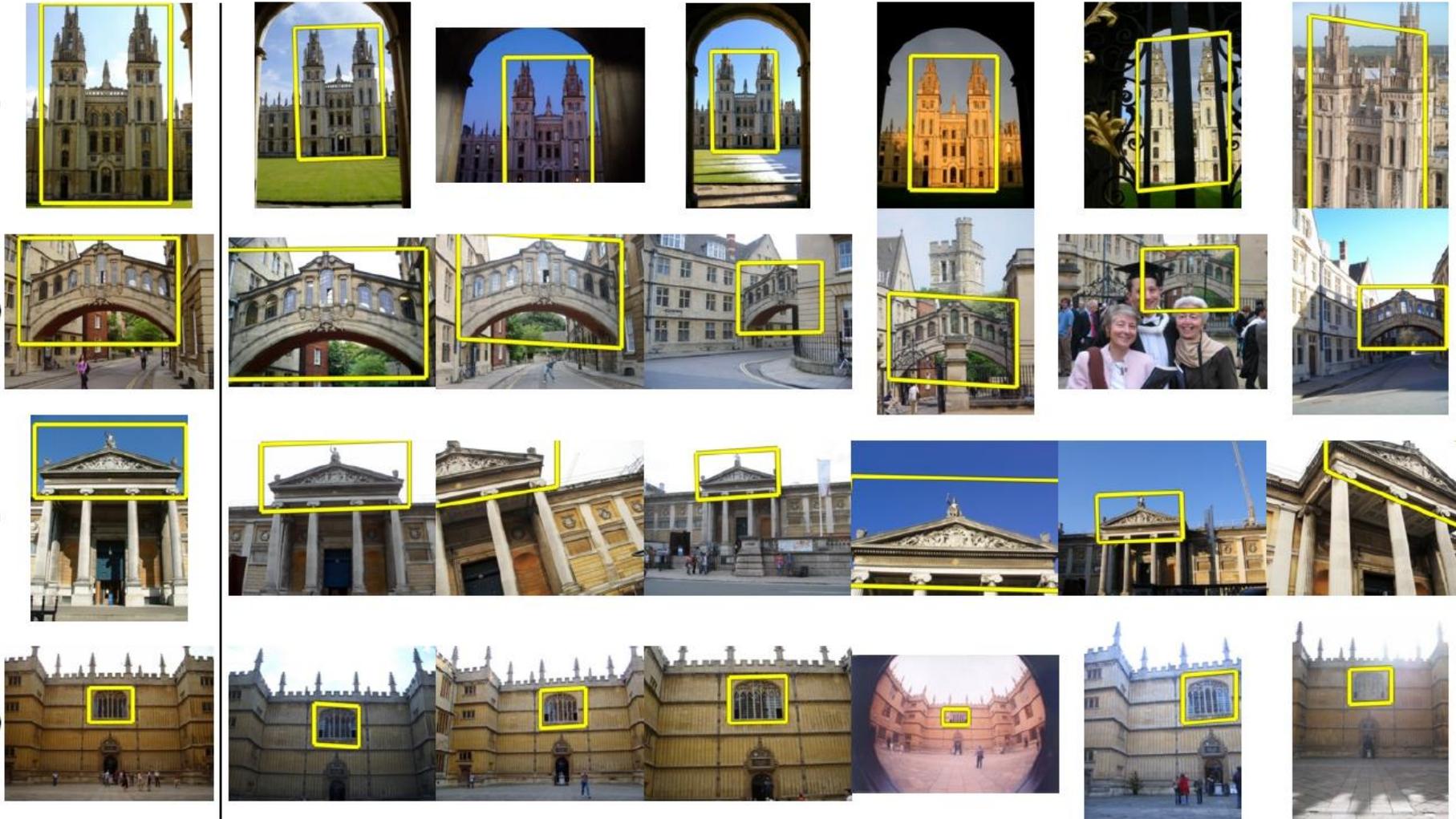


Query region



Retrieved frames

# Application: Large-Scale Retrieval



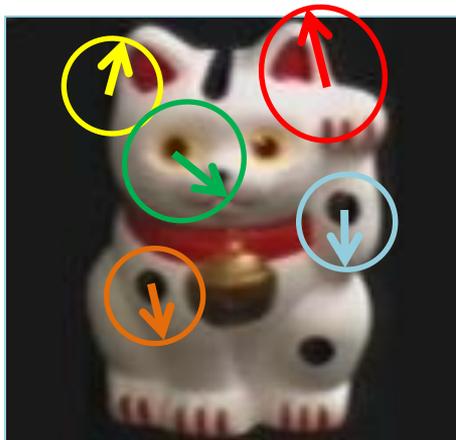
Query

Results from 5k Flickr images (demo available for 100k set)

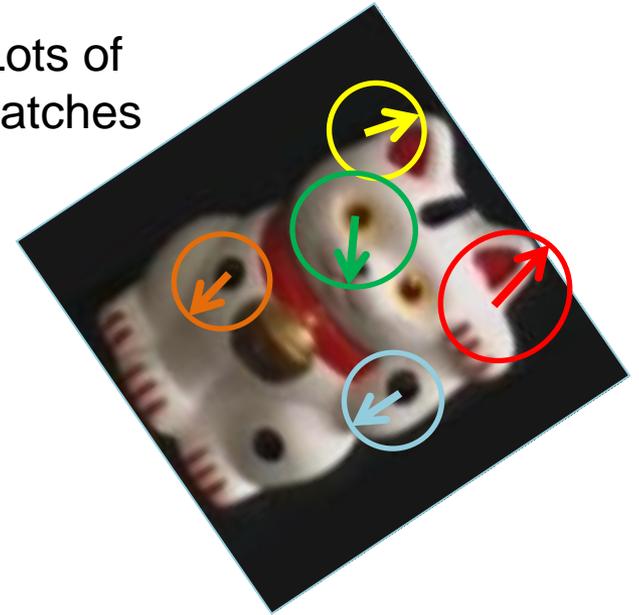
[Philbin CVPR'07]

# Simple idea

See how many keypoints are close to keypoints in each other image



Lots of Matches



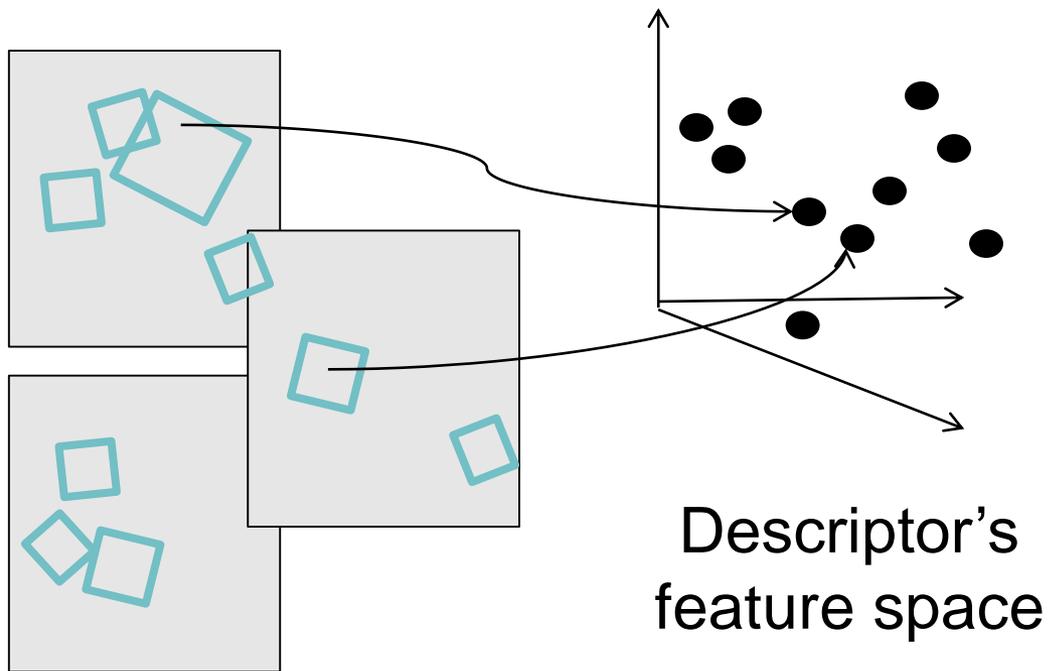
But this will be really, really slow!

Few or No Matches



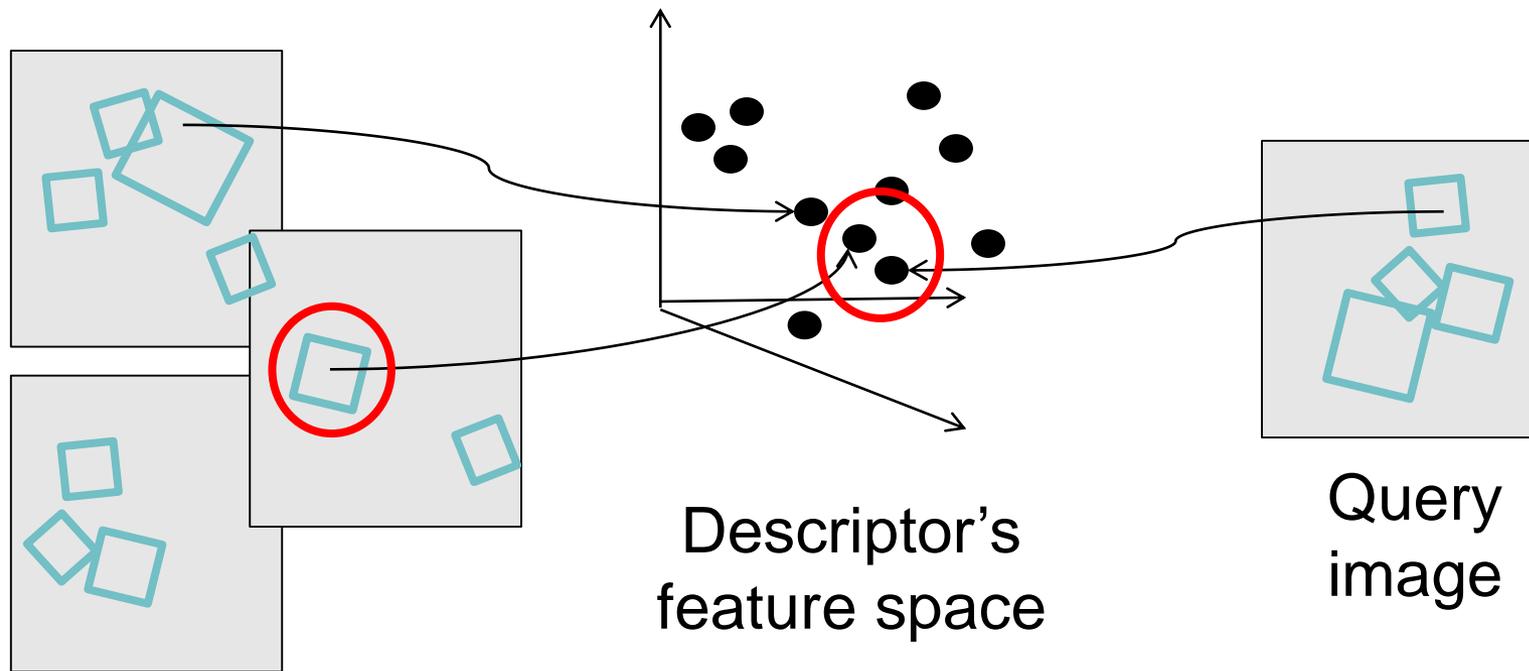
# Indexing local features

- Each patch / region has a descriptor, which is a point in some high-dimensional feature space (e.g., SIFT)



# Indexing local features

- When we see close points in feature space, we have similar descriptors, which indicates similar local content.



Database  
images

Descriptor's  
feature space

Query  
image

*Easily can have millions of  
features to search!*

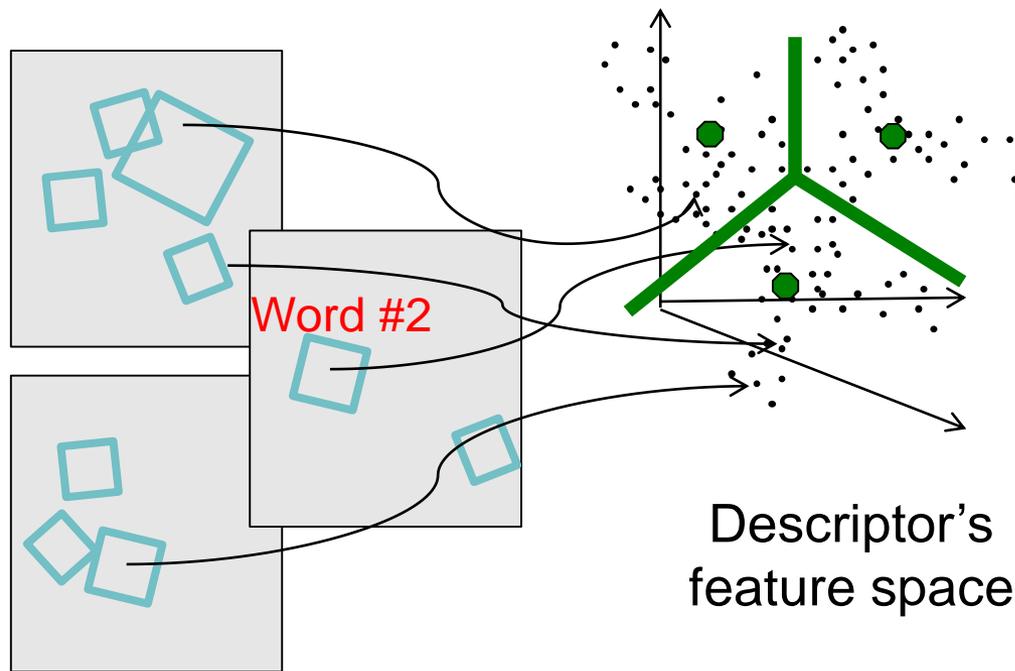
# Indexing local features: inverted file index

Index	
"Along I-75," From Detroit to Florida; <i>inside back cover</i>	Butterfly Center, McGuire; 134
"Drive I-95," From Boston to Florida; <i>inside back cover</i>	CAA (see AAA)
1929 Spanish Trail Roadway; 101-102,104	CCC, The; 111,113,115,135,142
511 Traffic Information; 83	Ca d'Zan; 147
A1A (Barrier Isl) - I-95 Access; 86	Caloosahatchee River; 152
AAA (and CAA); 83	Name; 150
AAA National Office; 88	Canaveral Natnl Seashore; 173
Abbreviations,	Cannon Creek Airpark; 130
Colored 25 mile Maps; cover	Canopy Road; 106,169
Exit Services; 196	Cape Canaveral; 174
Travelogue; 85	Castillo San Marcos; 169
Africa; 177	Cave Diving; 131
Agricultural Inspection Stns; 126	Cayo Costa, Name; 150
Ah-Tah-Thi-Ki Museum; 160	Celebration; 93
Air Conditioning, First; 112	Charlotte County; 149
Alabama; 124	Charlotte Harbor; 150
Alachua; 132	Chautauqua; 116
County; 131	ChIPLEY; 114
Alafia River; 143	Name; 115
Alapaha, Name; 126	Choctawatchee, Name; 115
Alfred B Maclay Gardens; 106	Circus Museum, Ringling; 147
Alligator Alley; 154-155	Citrus; 88,97,130,136,140,180
Alligator Farm, St Augustine; 169	CityPlace, W Palm Beach; 180
Alligator Hole (definition); 157	City Maps,
Alligator, Buddy; 155	Fl Lauderdale Expwys; 194-195
Alligators; 100,135,138,147,156	Jacksonville; 163
Anastasia Island; 170	Kissimmee Expwys; 192-193
Anhaica; 109-109,146	Miami Expressways; 194-195
Apalachicola River; 112	Orlando Expressways; 192-193
Appleton Mus of Art; 136	Pensacola; 26
Aquifer; 102	Tallahassee; 191
Arabian Nights; 94	Tampa-St. Petersburg; 63
Art Museum, Ringling; 147	St. Augustine; 191
Aruba Beach Cafe; 183	Civil War; 100,108,127,138,141
Aucilla River Project; 106	Clearwater Marine Aquarium; 187
Babcock-Web WMA; 151	Collier County; 154
Bahia Mar Marina; 184	Collier, Barron; 152
Baker County; 99	Colonial Spanish Quarters; 168
Barefoot Mailmen; 182	Columbia County; 101,128
Barge Canal; 137	Coquina Building Material; 165
Bee Line Expy; 80	Corkscrew Swamp, Name; 154
Belz Outlet Mall; 89	Cowboys; 85
Bernard Castro; 136	Crab Trap II; 144
Big "I"; 165	Cracker, Florida; 88,95,132
Big Cypress; 155,158	Crosstown Expy; 11,35,98,143
Big Foot Monster; 105	Cuban Bread; 184
Billie Swamp Safari; 160	Dade Battlefield; 140
Blackwater River SP; 117	Dade, Maj. Francis; 139-140,161
Blue Angels	Dania Beach Hurricane; 184
	Daniel Boone, Florida Walk; 117
	Daytona Beach; 172-173
	De Land; 87
	Driving Lanes; 85
	Duval County; 163
	Eau Gallie; 175
	Edison, Thomas; 152
	Eglin AFB; 116-118
	Eight Reale; 176
	Ellenton; 144-145
	Emanuel Point Wreck; 120
	Emergency Callboxes; 83
	Epiphytes; 142,148,157,159
	Escambia Bay; 119
	Bridge (I-10); 119
	County; 120
	Estero; 153
	Everglade,90,95,139-140,154-160
	Draining of; 156,181
	Wildlife MA; 160
	Wonder Gardens; 154
	Falling Waters SP; 115
	Fantasy of Flight; 95
	Fayer Dykes SP; 171
	Fires, Forest; 168
	Fires, Prescribed ; 148
	Fisherman's Village; 151
	Flagler County; 171
	Flagler, Henry; 97,165,167,171
	Florida Aquarium; 186
	Florida,
	12,000 years ago; 187
	Cavern SP; 114
	Map of all Expressways; 2-3
	Mus of Natural History; 134
	National Cemetery ; 141
	Part of Africa; 177
	Platform; 187
	Sheriff's Boys Camp; 126
	Sports Hall of Fame; 130
	Sun 'n Fun Museum; 97
	Supreme Court; 107
	Florida's Turnpike (FTP), 178,189
	25 mile Strip Maps; 66
	Administration; 189
	Coin System; 190
	Exit Services; 189
	HEFT; 76,161,190
	History; 189
	Names; 189
	Service Plazas; 190
	Spur SR91; 76
	Ticket System; 190
	Toll Plazas; 190
	Ford, Henry; 152

- For text documents, an efficient way to find all *pages* on which a *word* occurs is to use an index...
- We want to find all *images* in which a *feature* occurs.
- To use this idea, we'll need to map our features to "visual words".

# Visual words

- Map high-dimensional descriptors to tokens/words by quantizing the feature space



- Quantize via clustering, let cluster centers be the prototype “words”
- Determine which word to assign to each new image region by finding the closest cluster center.

# Visual words

- Example: each group of patches belongs to the same visual word

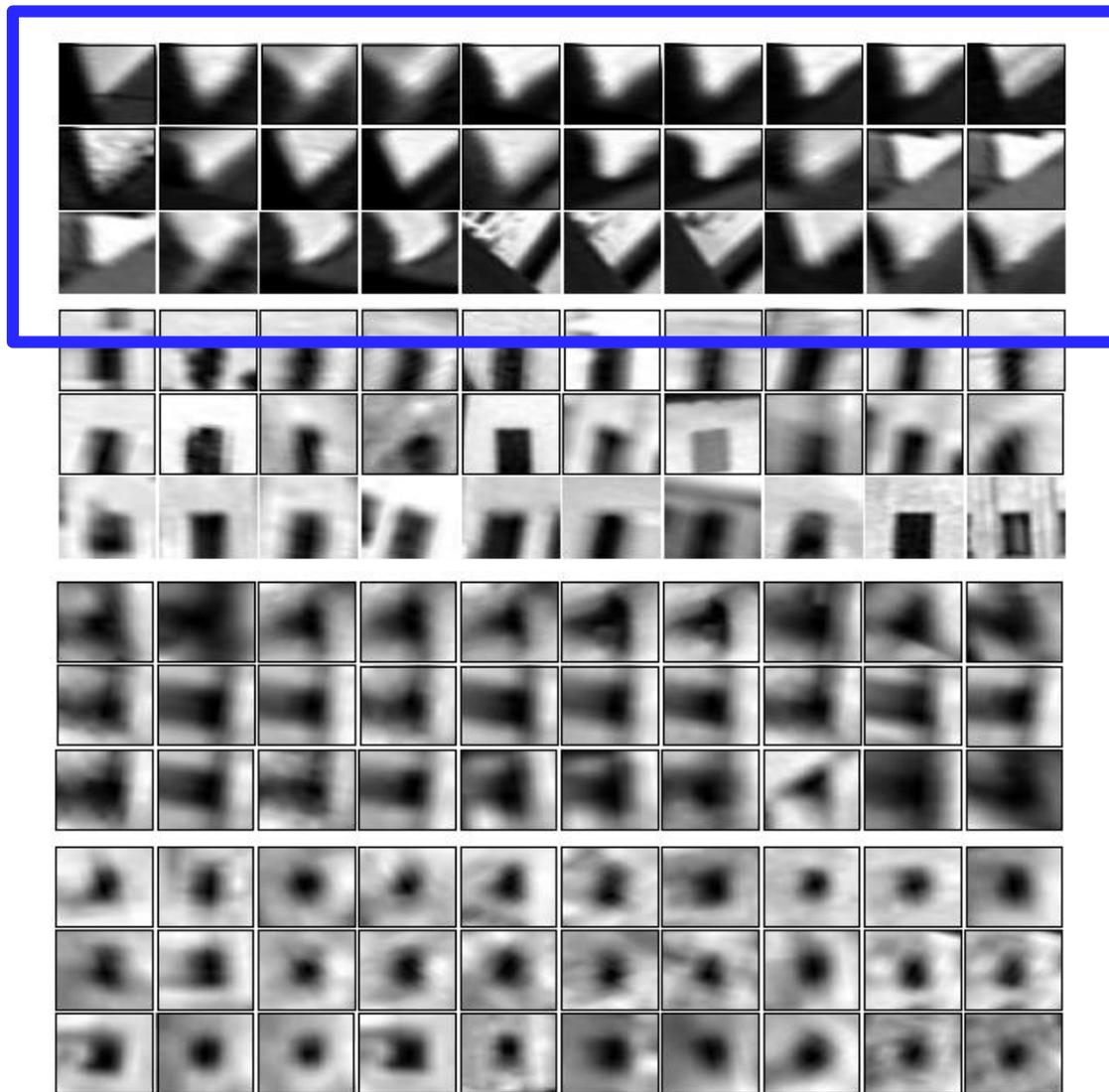
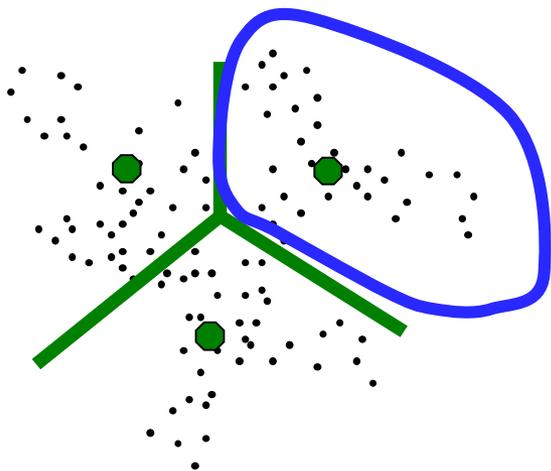


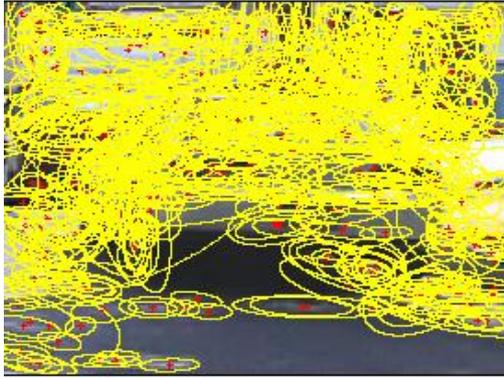
Figure from Sivic & Zisserman, ICCV 2003 Kristen Grauman

# Visual vocabulary formation

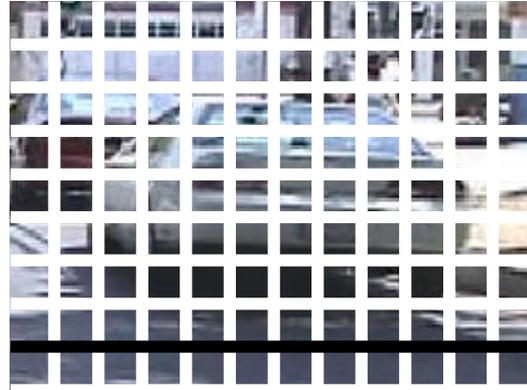
## Issues:

- Vocabulary size, number of words
- Sampling strategy: where to extract features?
- Clustering / quantization algorithm
- Unsupervised vs. supervised
- What corpus provides features (universal vocabulary?)

# Sampling strategies



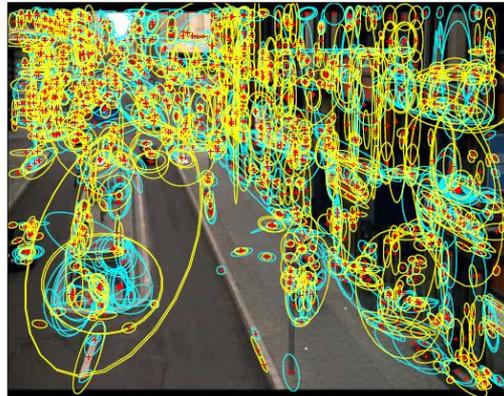
Sparse, at interest points



Dense, uniformly



Randomly



Multiple interest operators

- To find specific, textured objects, sparse sampling from interest points often more reliable.
- Multiple complementary interest operators offer more image coverage.
- For object categorization, dense sampling offers better coverage.

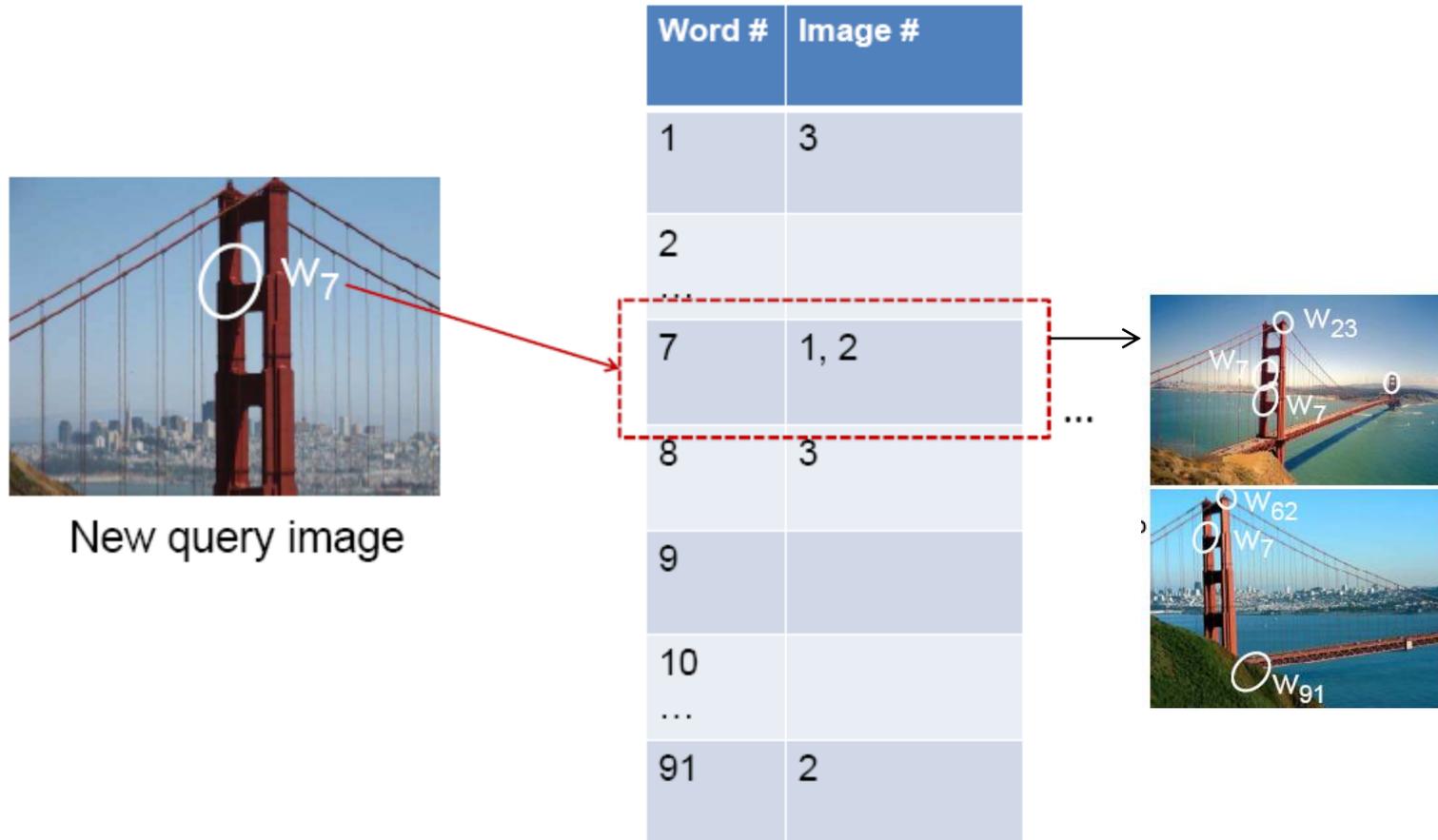
[See Nowak, Jurie & Triggs, ECCV 2006]

# Inverted file index



- Database images are loaded into the index mapping words to image numbers

# Inverted file index



- New query image is mapped to indices of database images that share a word.

# Inverted file index

- Key requirement for inverted file index to be efficient: sparsity
- If most pages/images contain most words then you're no better off than exhaustive search.
  - Exhaustive search would mean comparing the word distribution of a query versus every page.

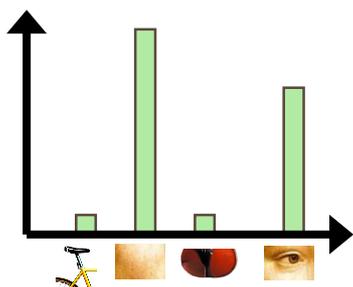
# Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

# Comparing bags of words

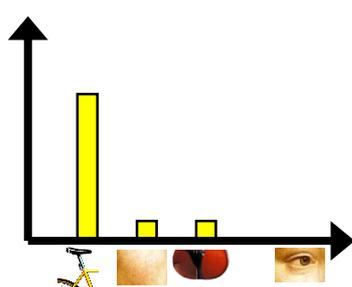
- Rank frames by normalized scalar product between their (possibly weighted) occurrence counts---*nearest neighbor* search for similar images.

[1 8 1 4]



$\vec{d}_j$

[5 1 1 0]



$\vec{q}$

$$\text{sim}(d_j, q) = \frac{\langle d_j, q \rangle}{\|d_j\| \|q\|}$$

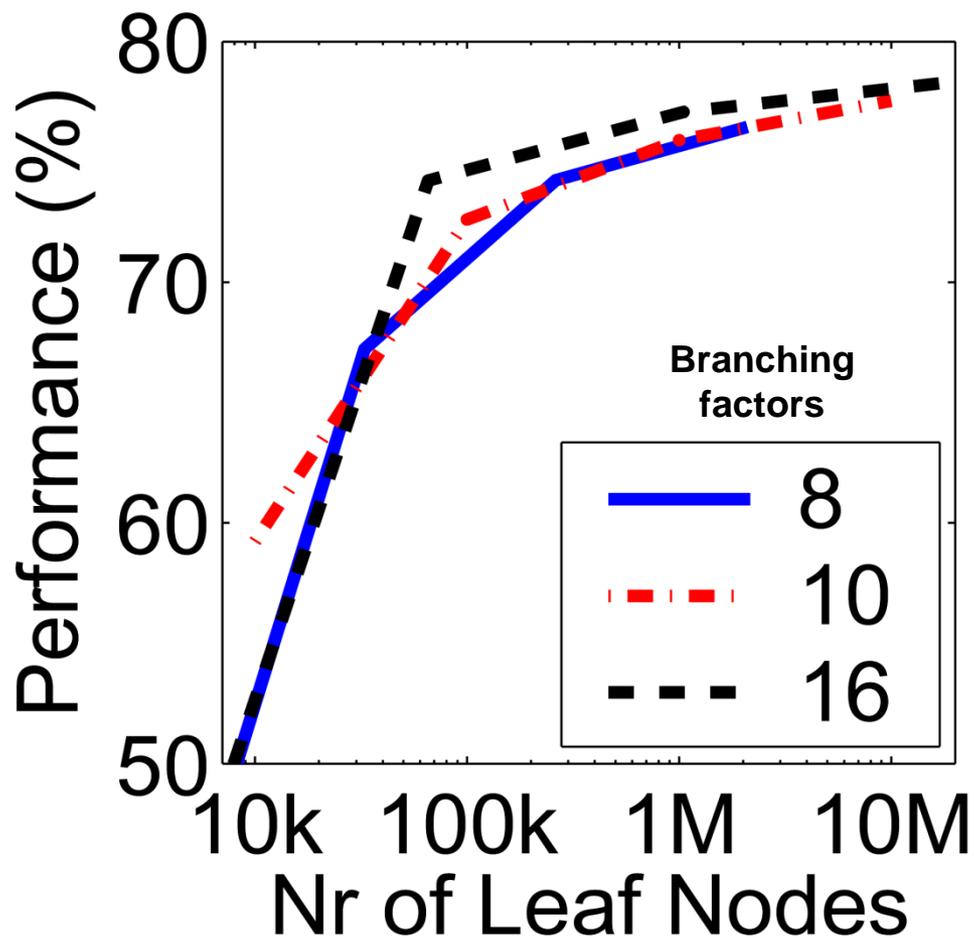
$$= \frac{\sum_{i=1}^V d_j(i) * q(i)}{\sqrt{\sum_{i=1}^V d_j(i)^2} * \sqrt{\sum_{i=1}^V q(i)^2}}$$

for vocabulary of  $V$  words

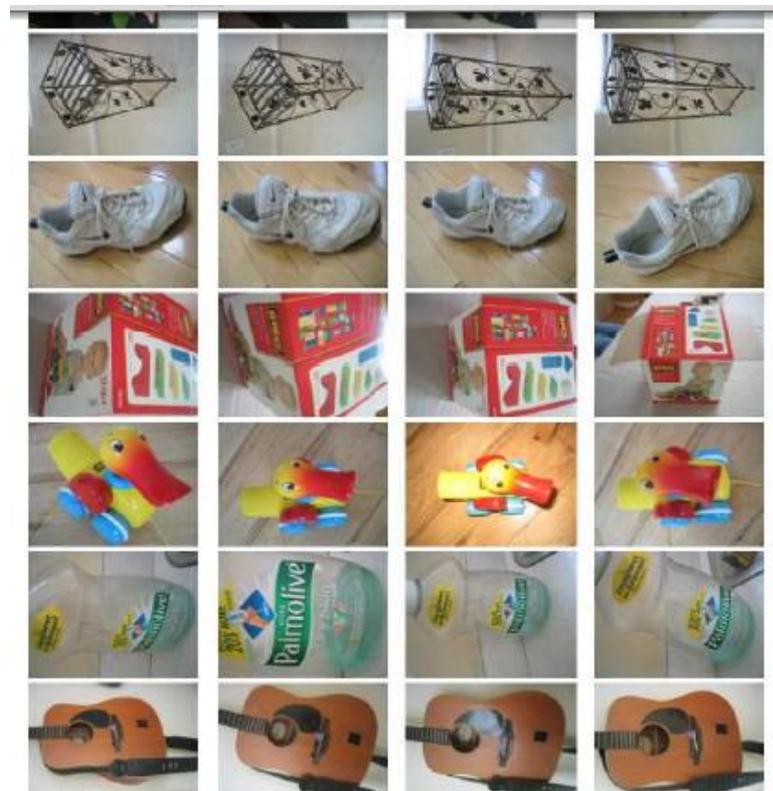
# Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

# Vocabulary size



Results for recognition task with 6347 images

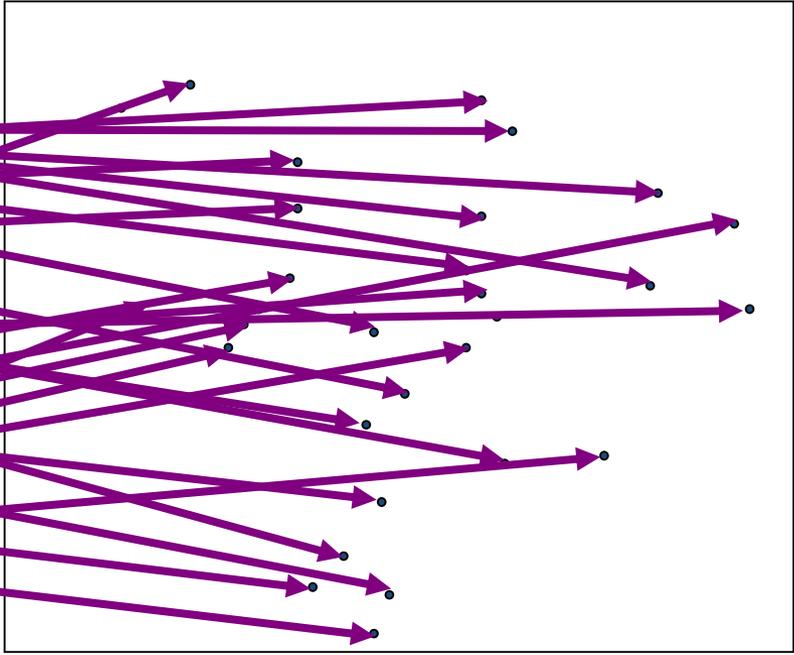
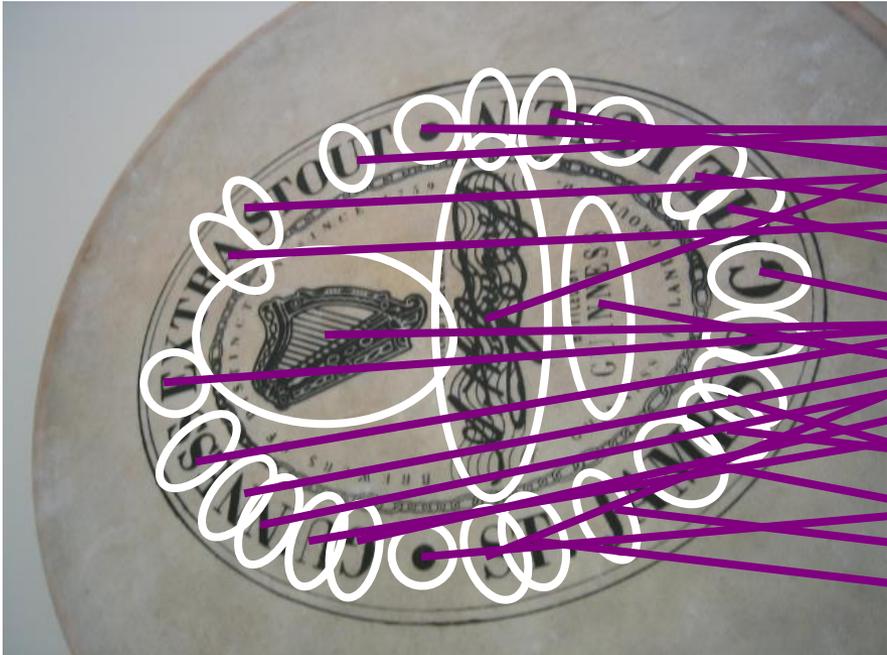


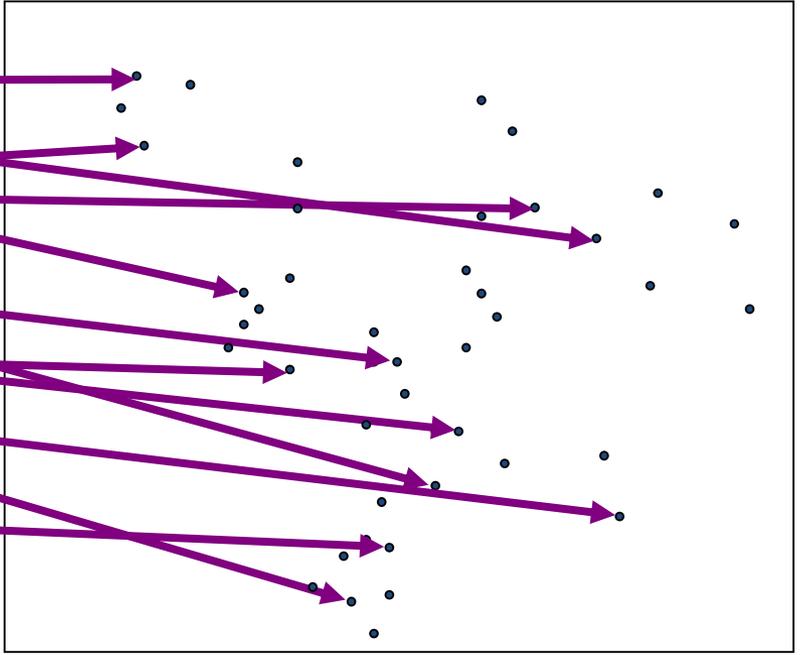
*Influence on performance, sparsity*

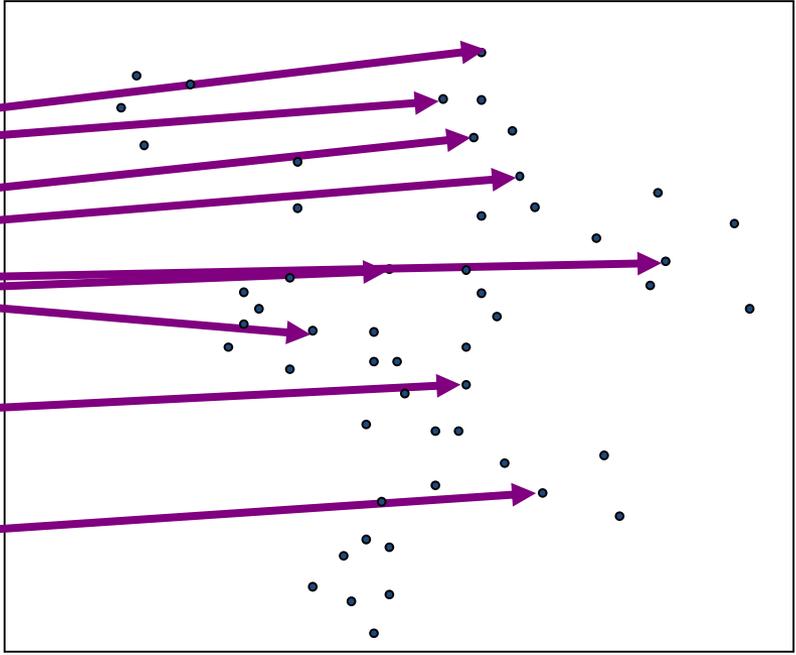
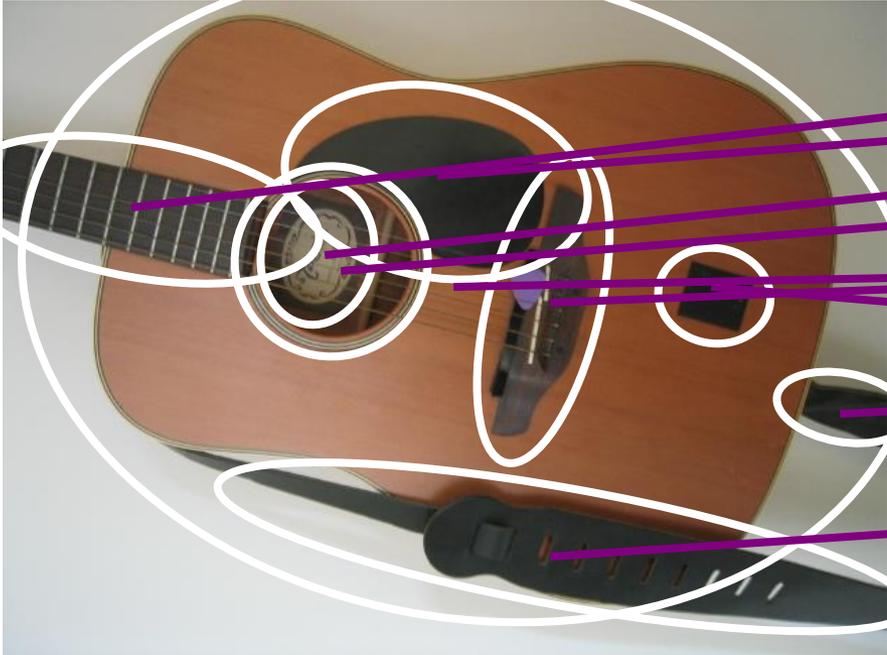
Nister & Stewenius, CVPR 2006  
Kristen Grauman

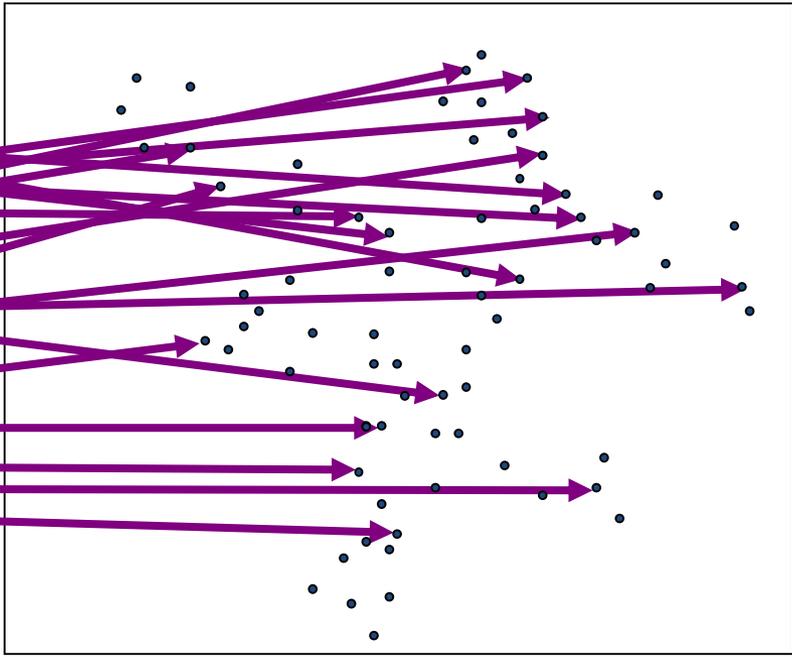
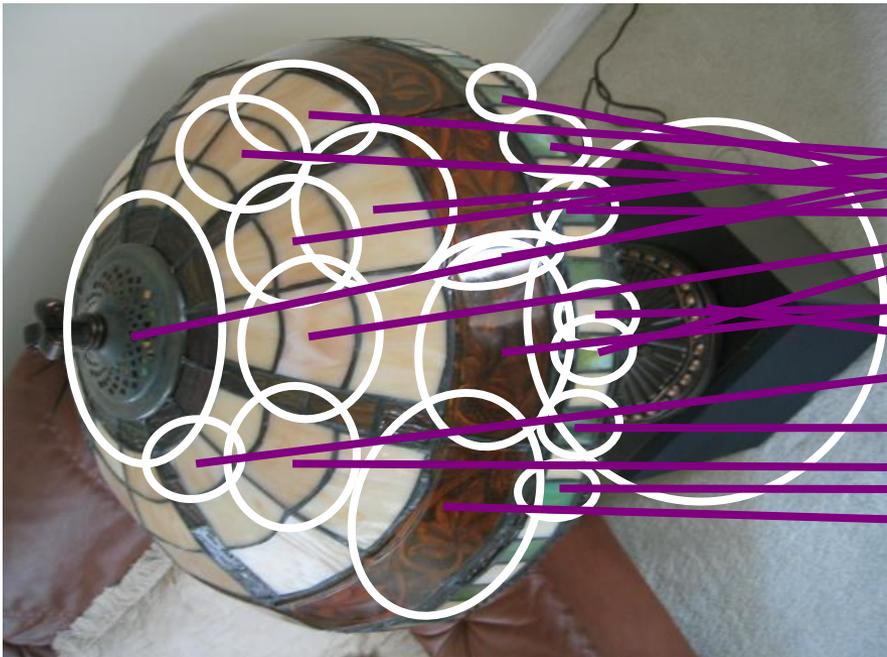
# Recognition with K-tree

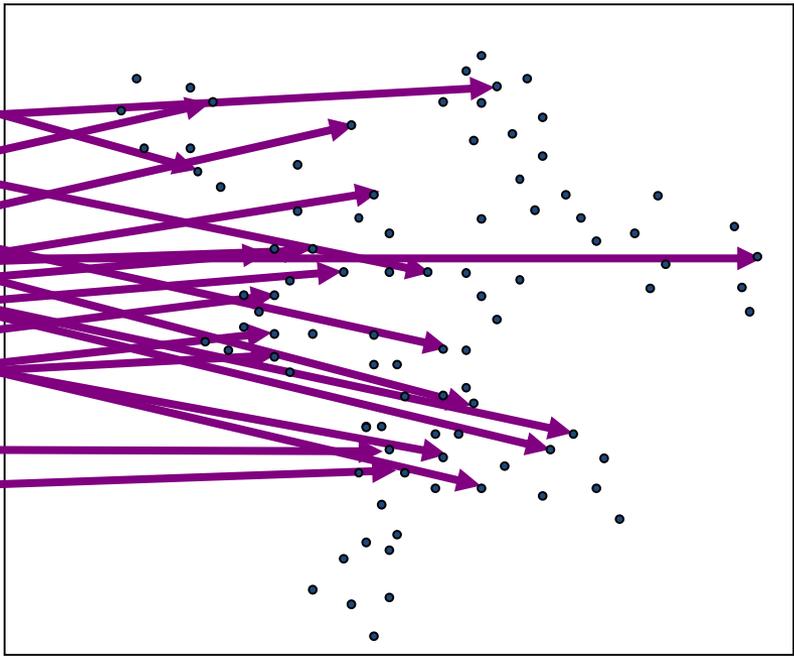
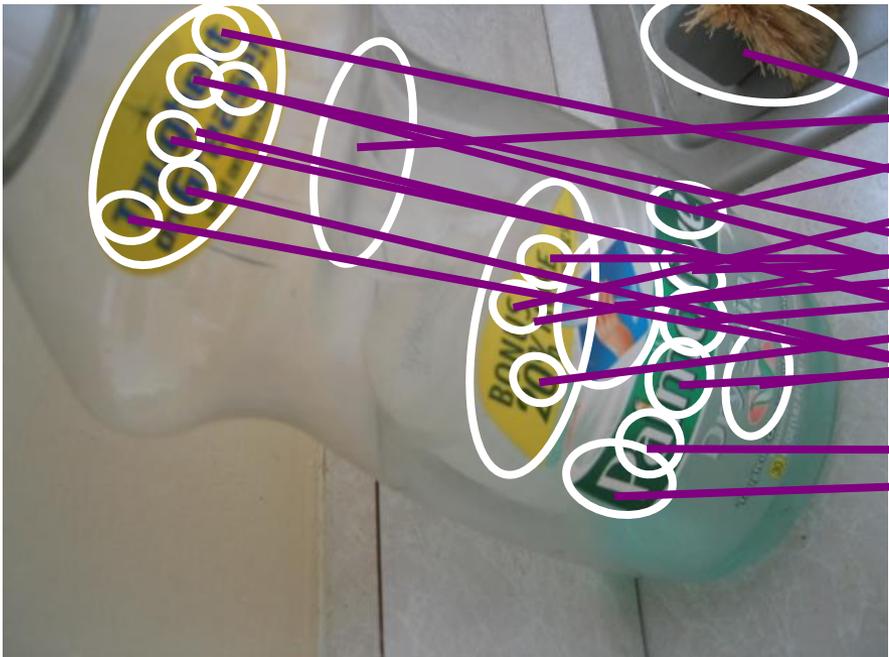
Following slides by David Nister (CVPR 2006)

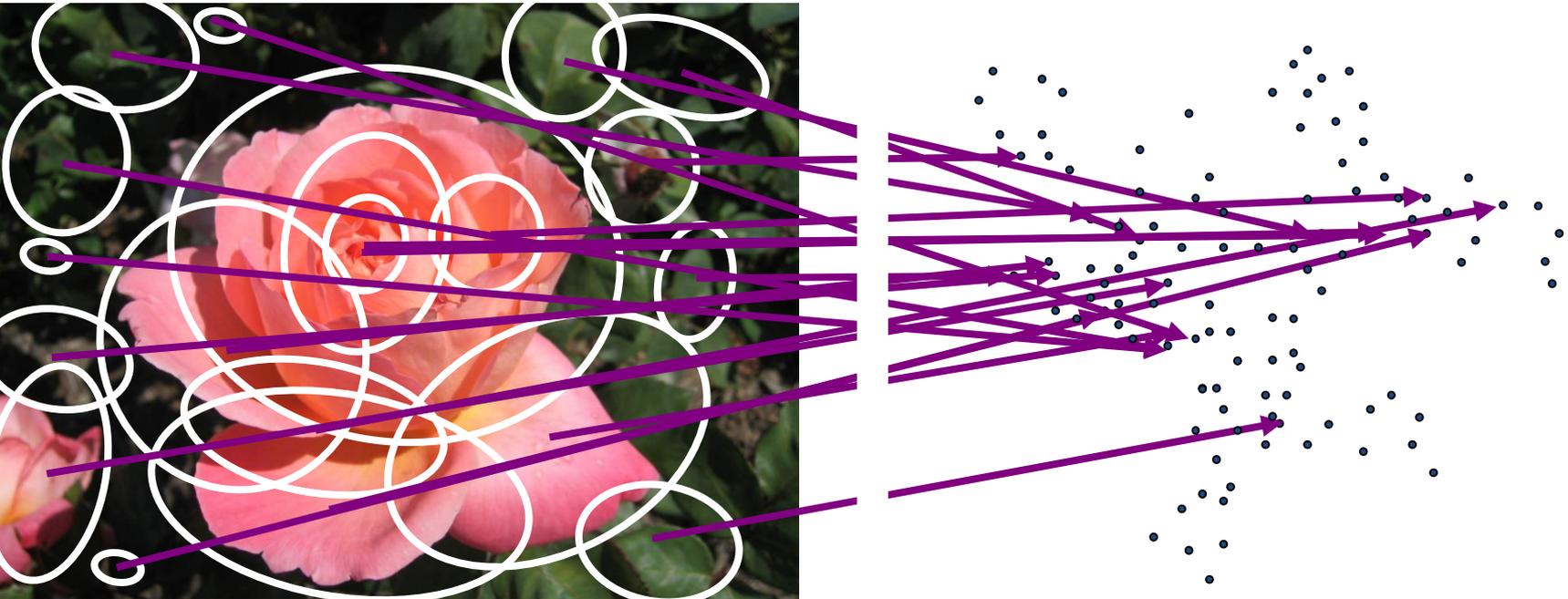


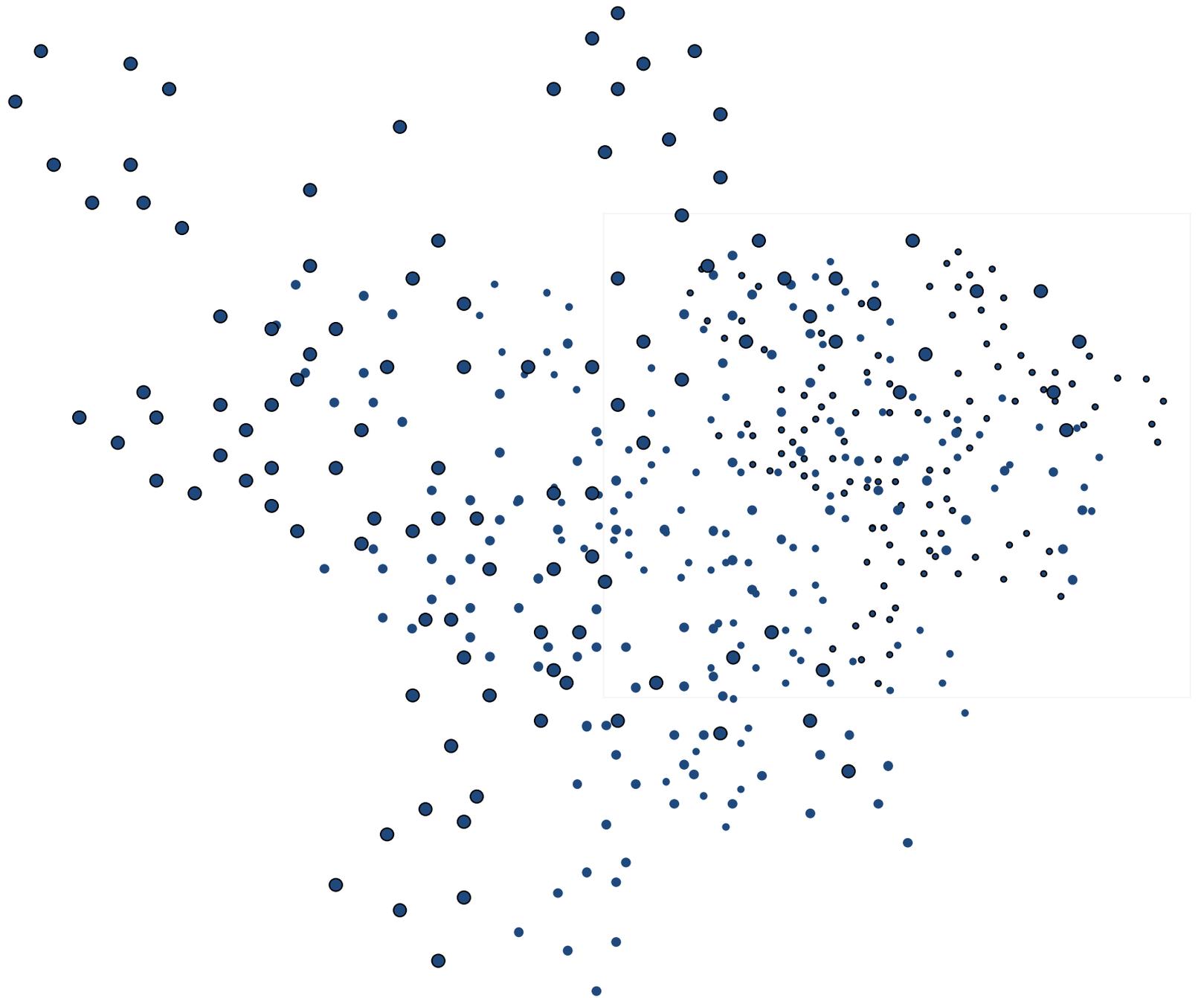


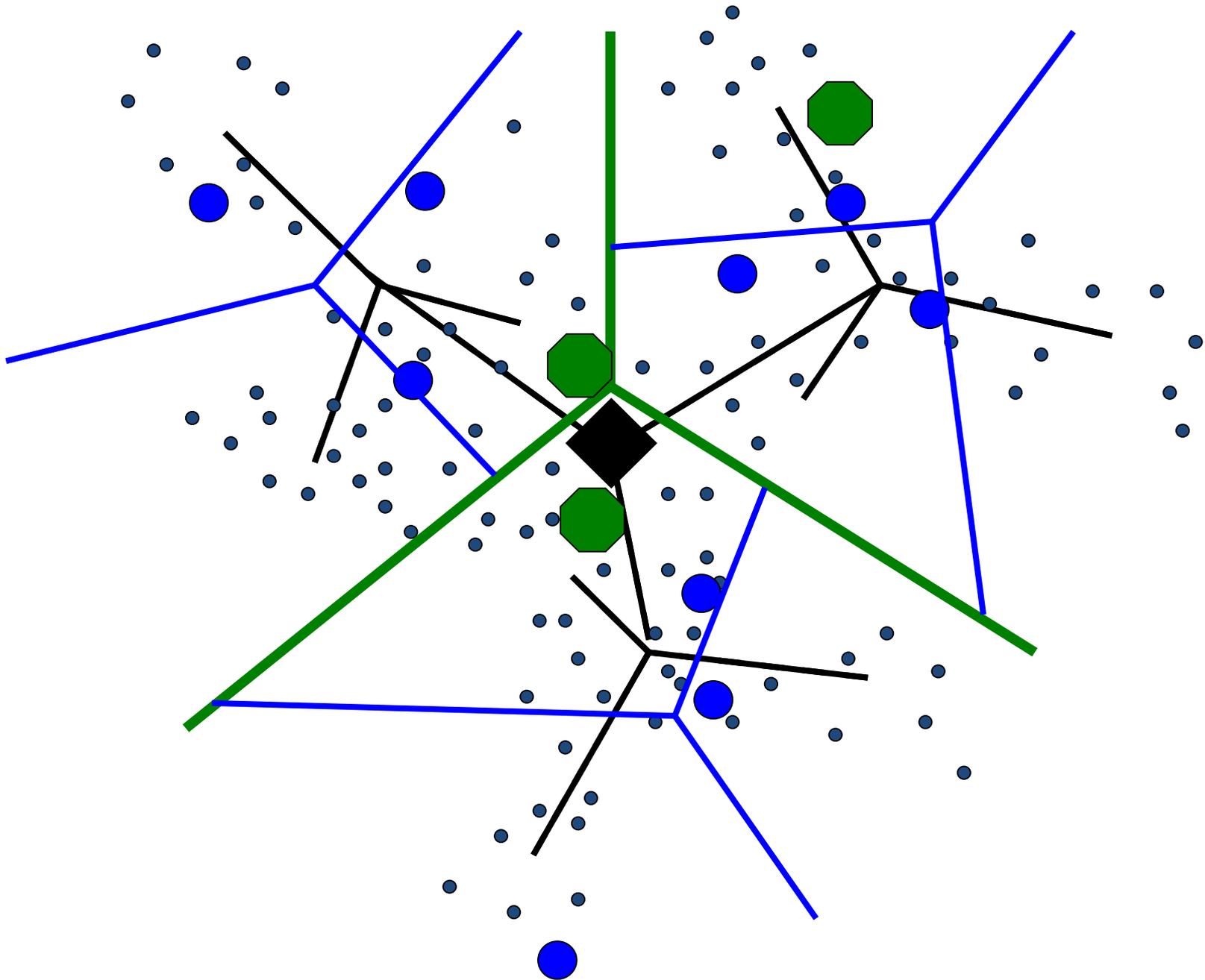


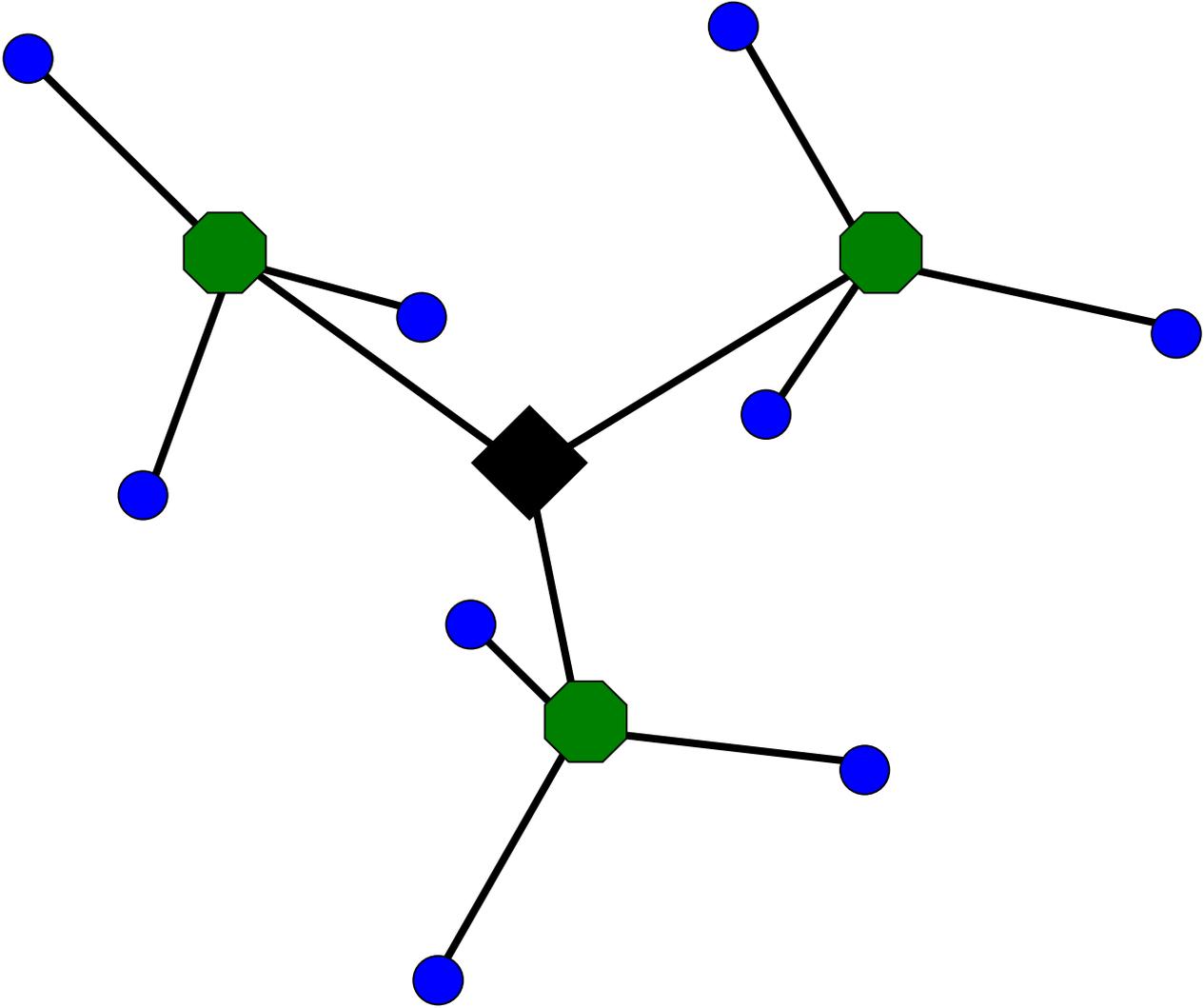


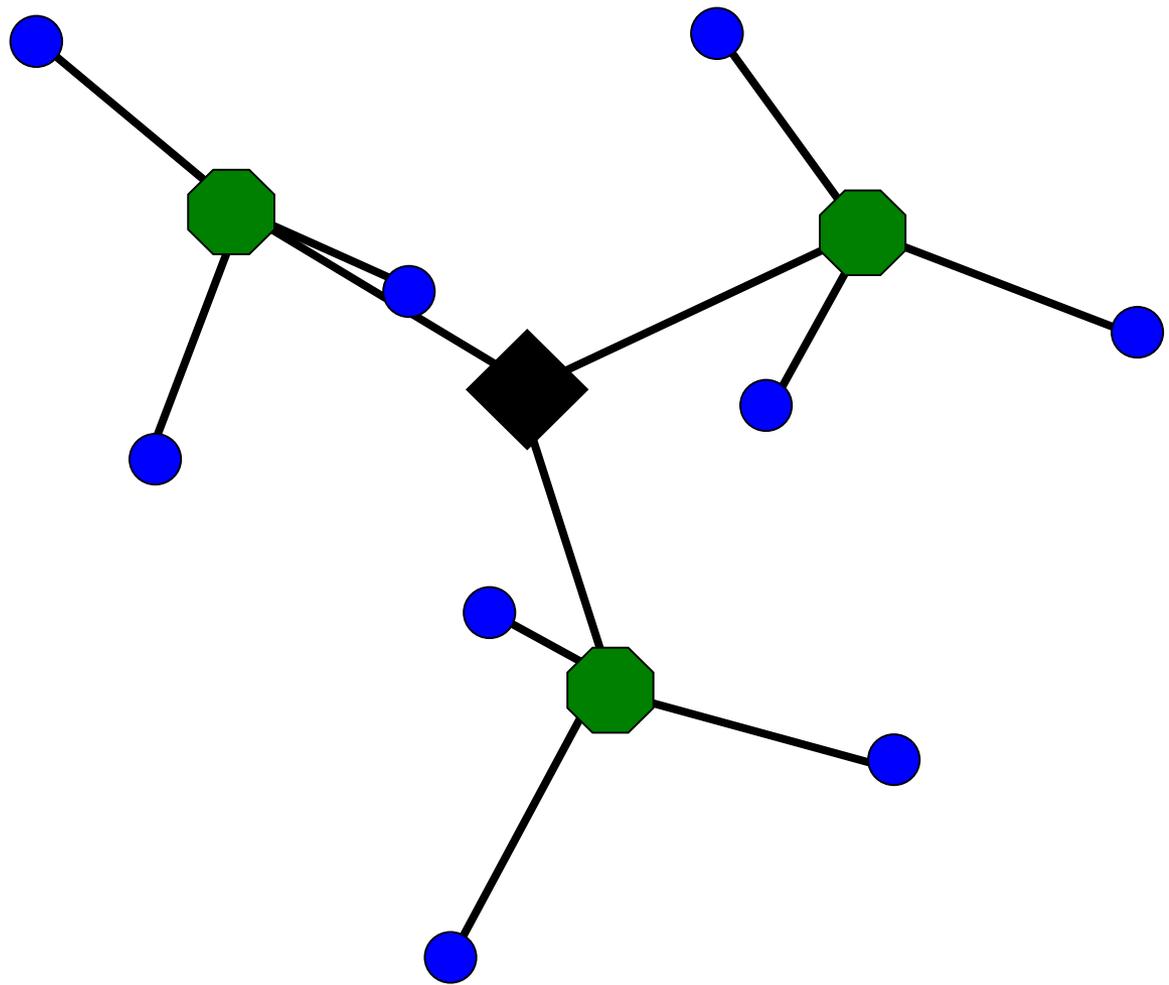


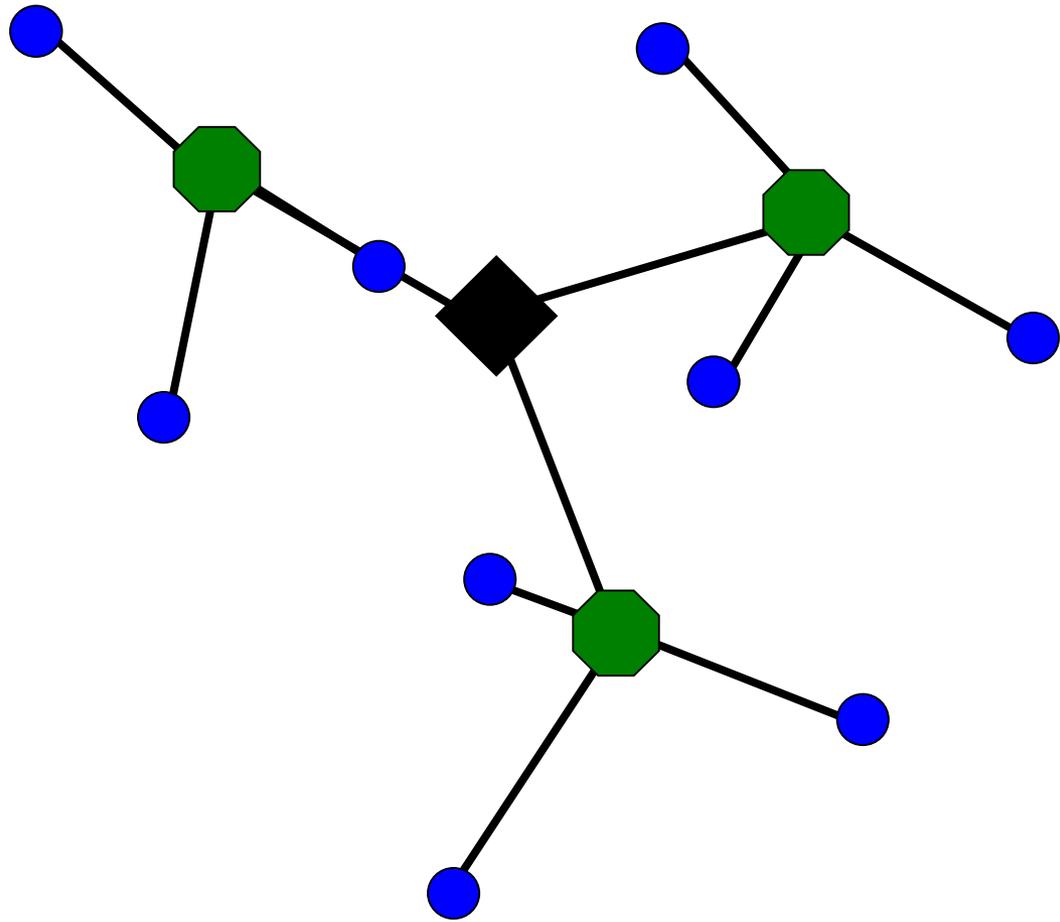


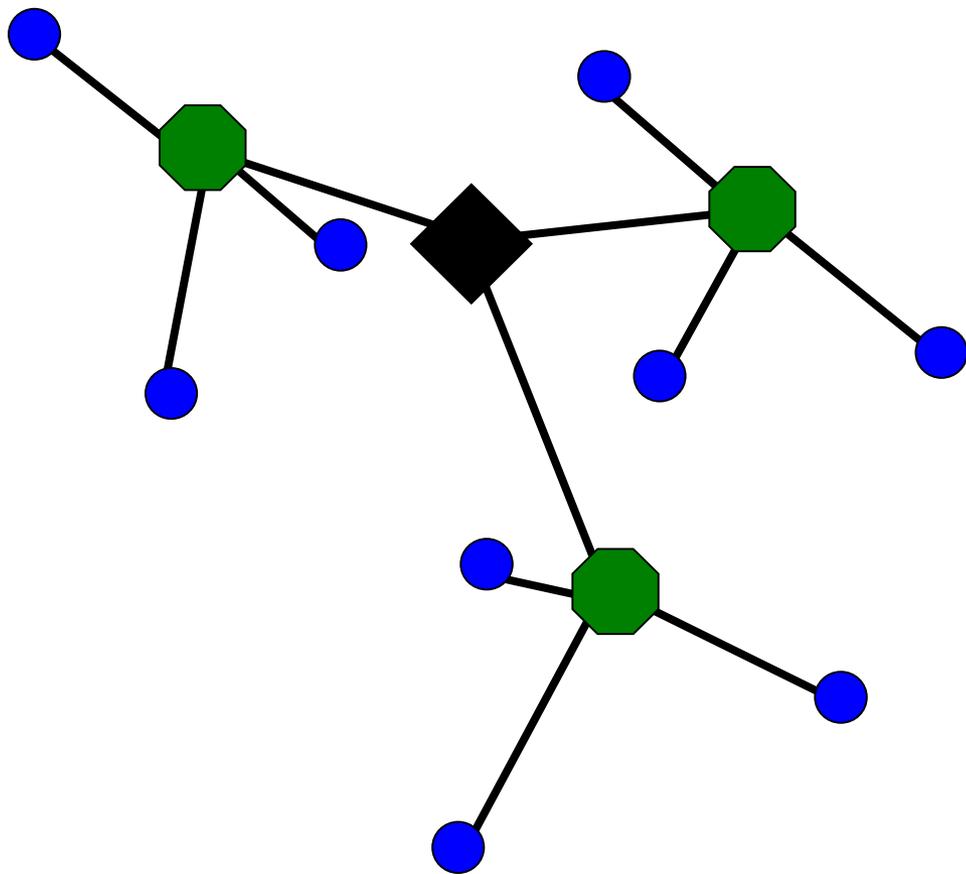


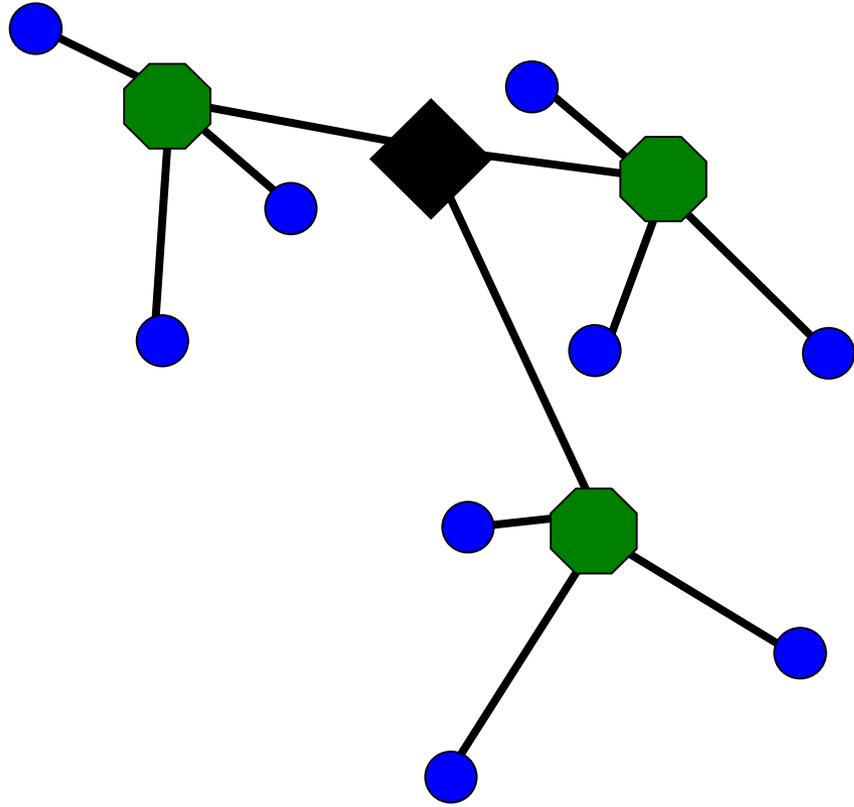


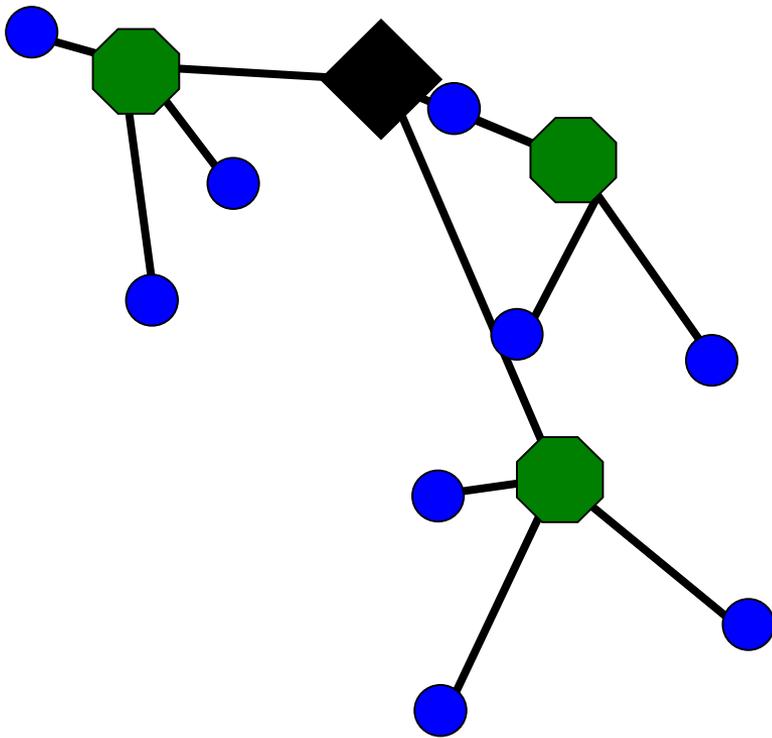


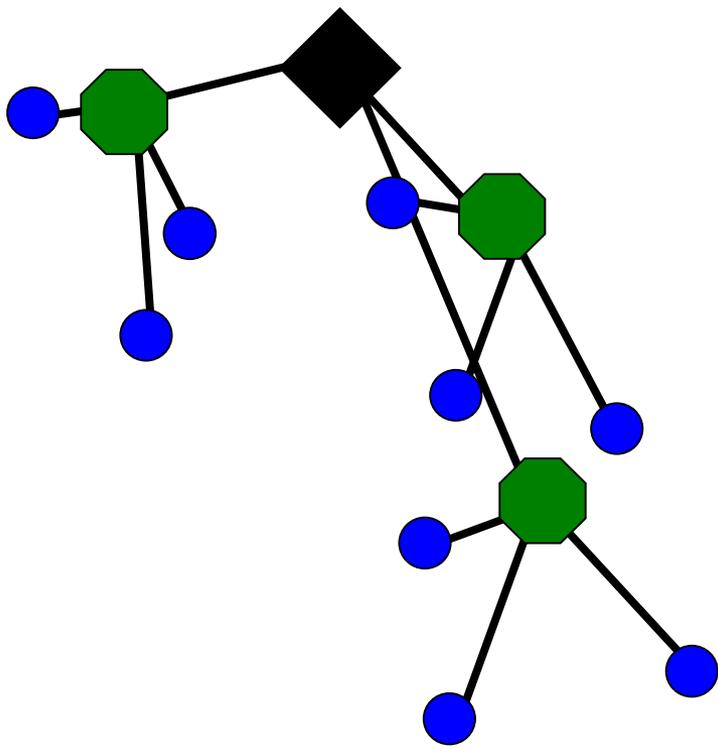


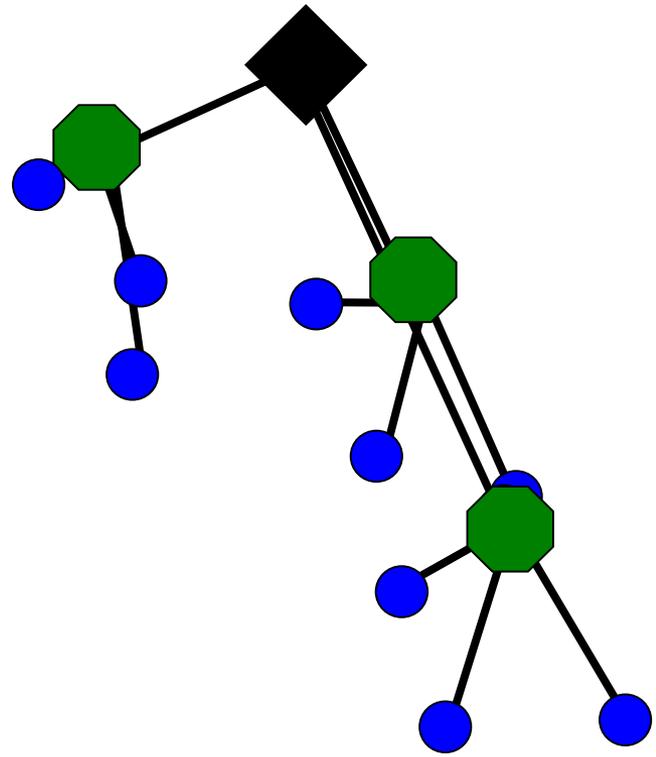


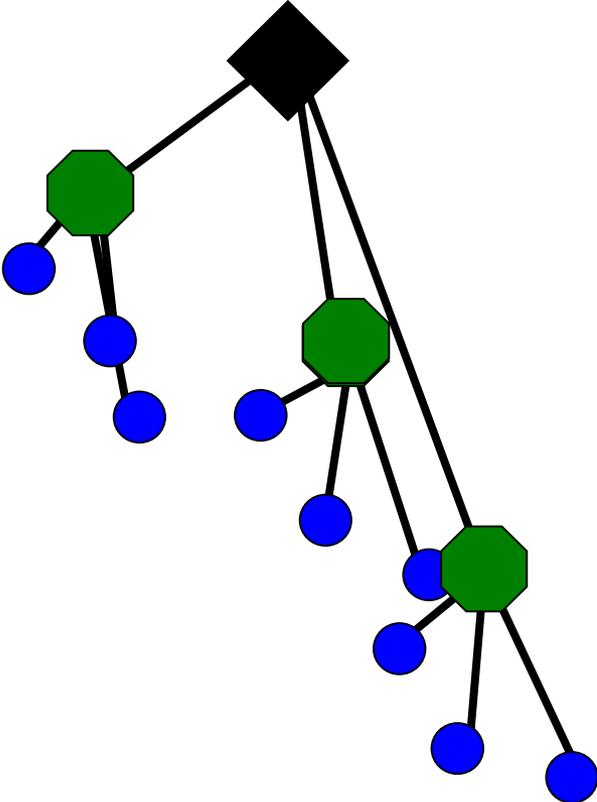


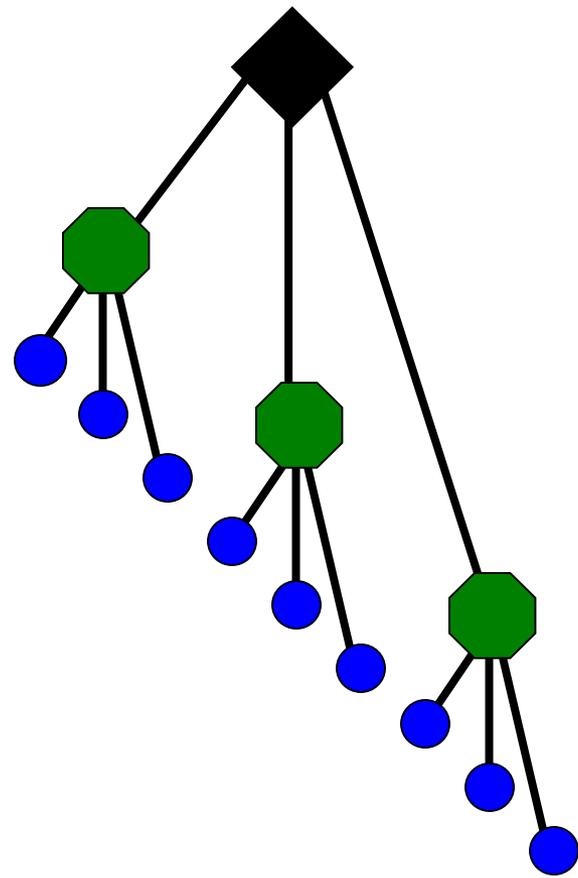


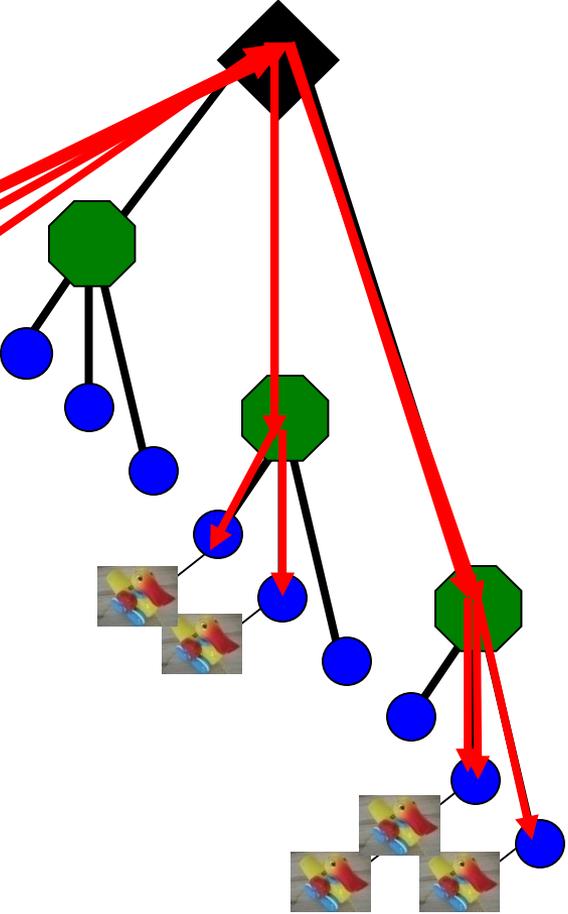
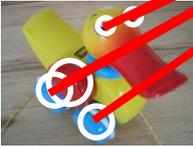


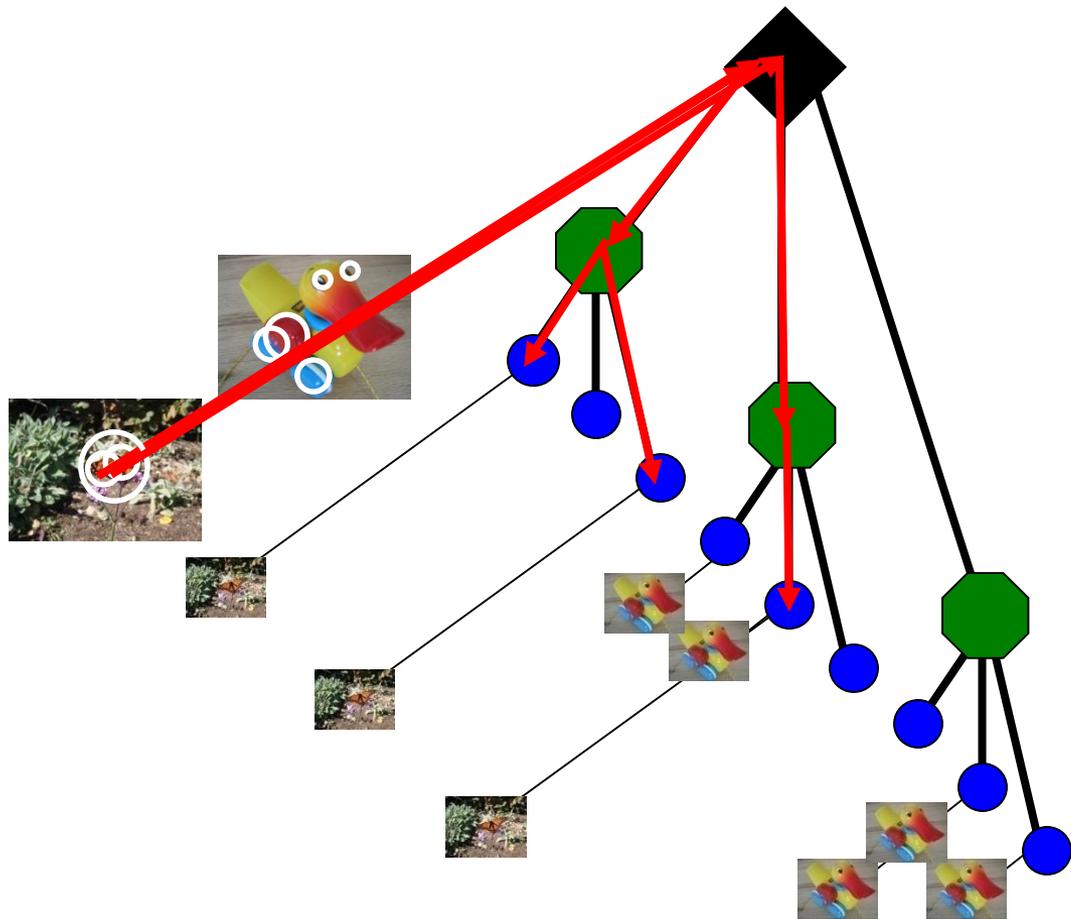


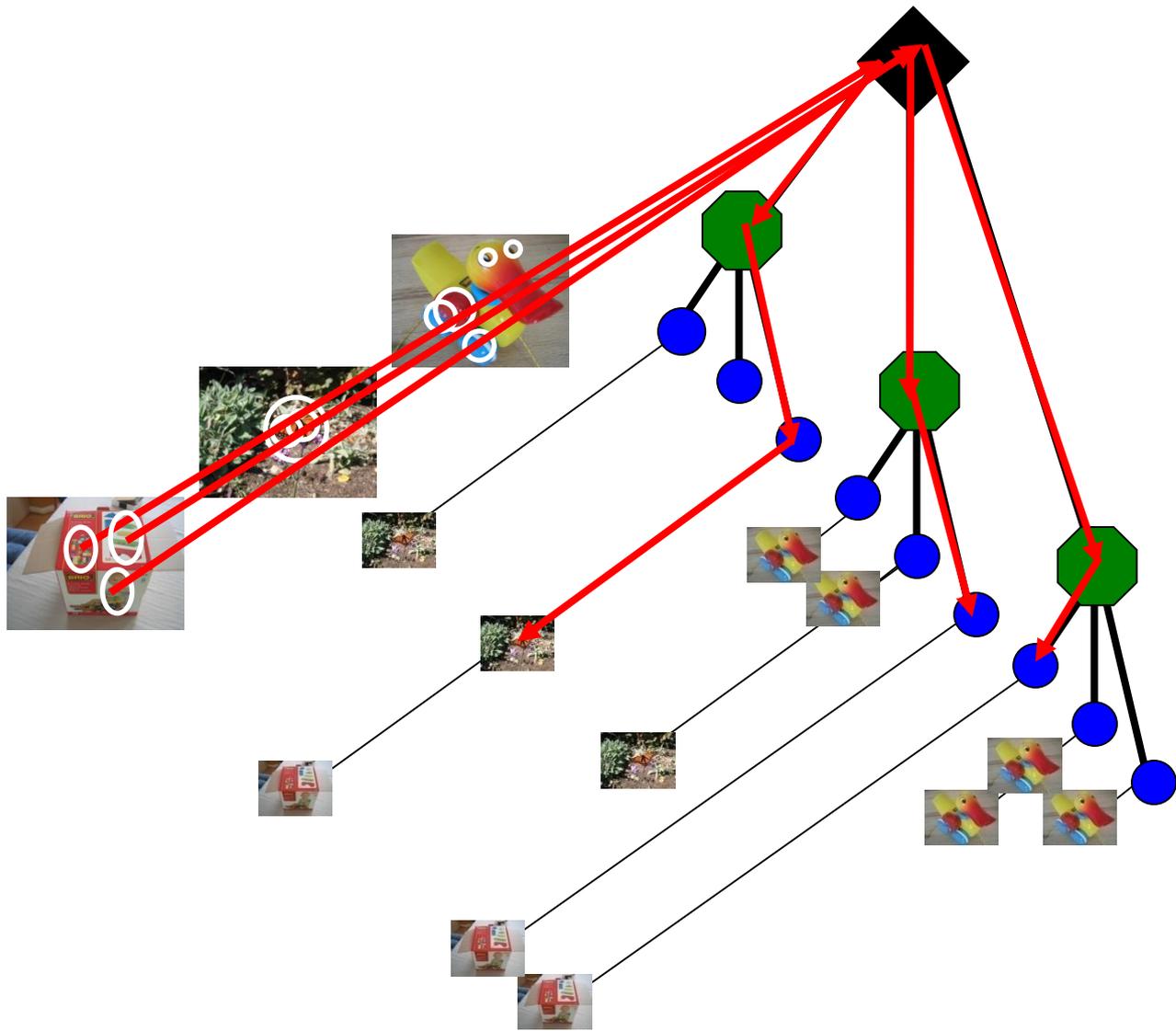




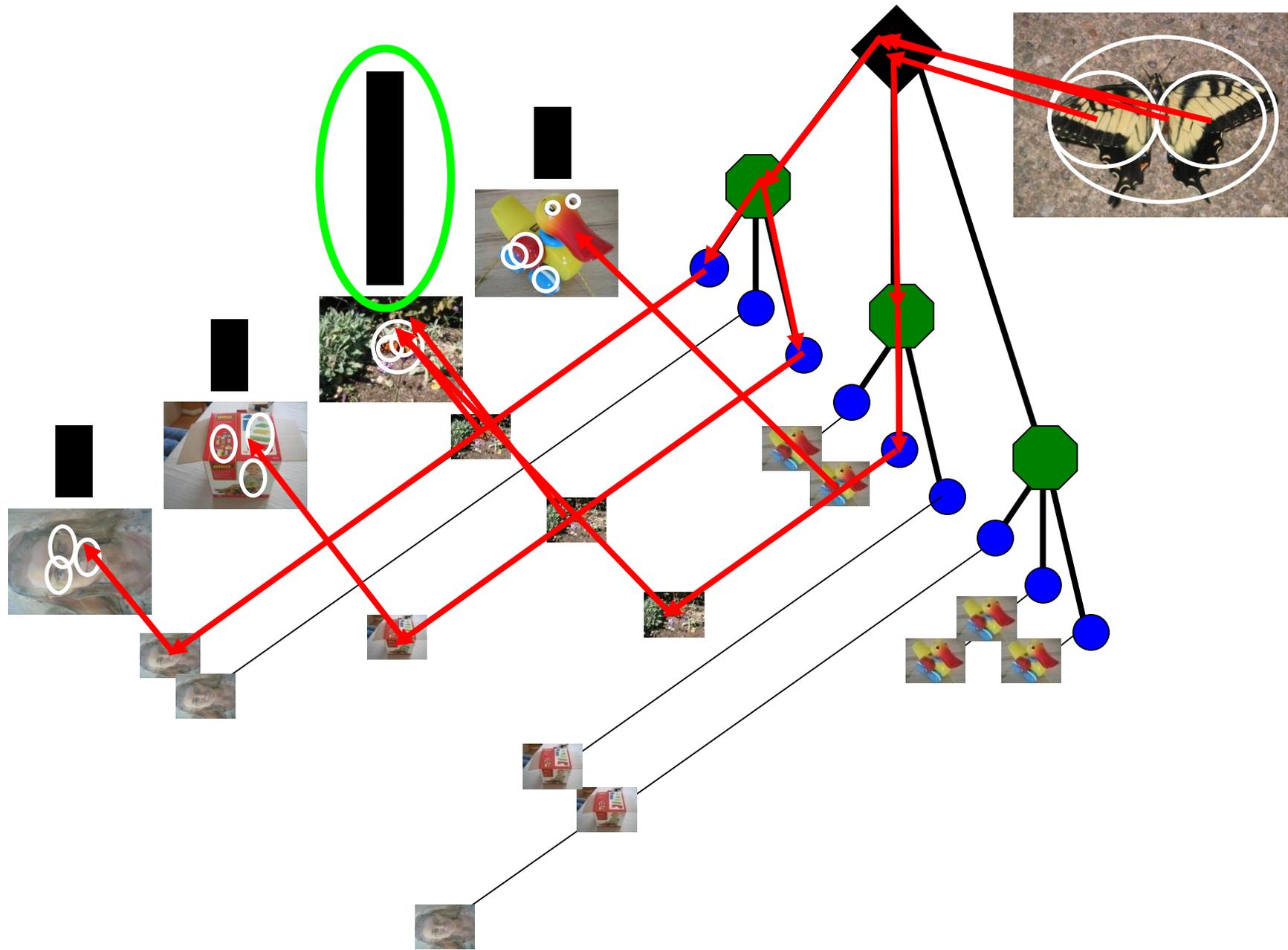












# Vocabulary trees: complexity

Number of words given tree parameters: branching factor and number of levels

$$\text{branching\_factor}^{\text{number\_of\_levels}}$$

Word assignment cost vs. flat vocabulary

$O(k)$  for flat

$O(\log_{\text{branching\_factor}}(k) * \text{branching\_factor})$

Is this like a kd-tree?

Yes, but with better partitioning and defeatist search.

This hierarchical data structure is lossy – you might not find your true nearest cluster.

110,000,000  
Images in  
5.8 Seconds

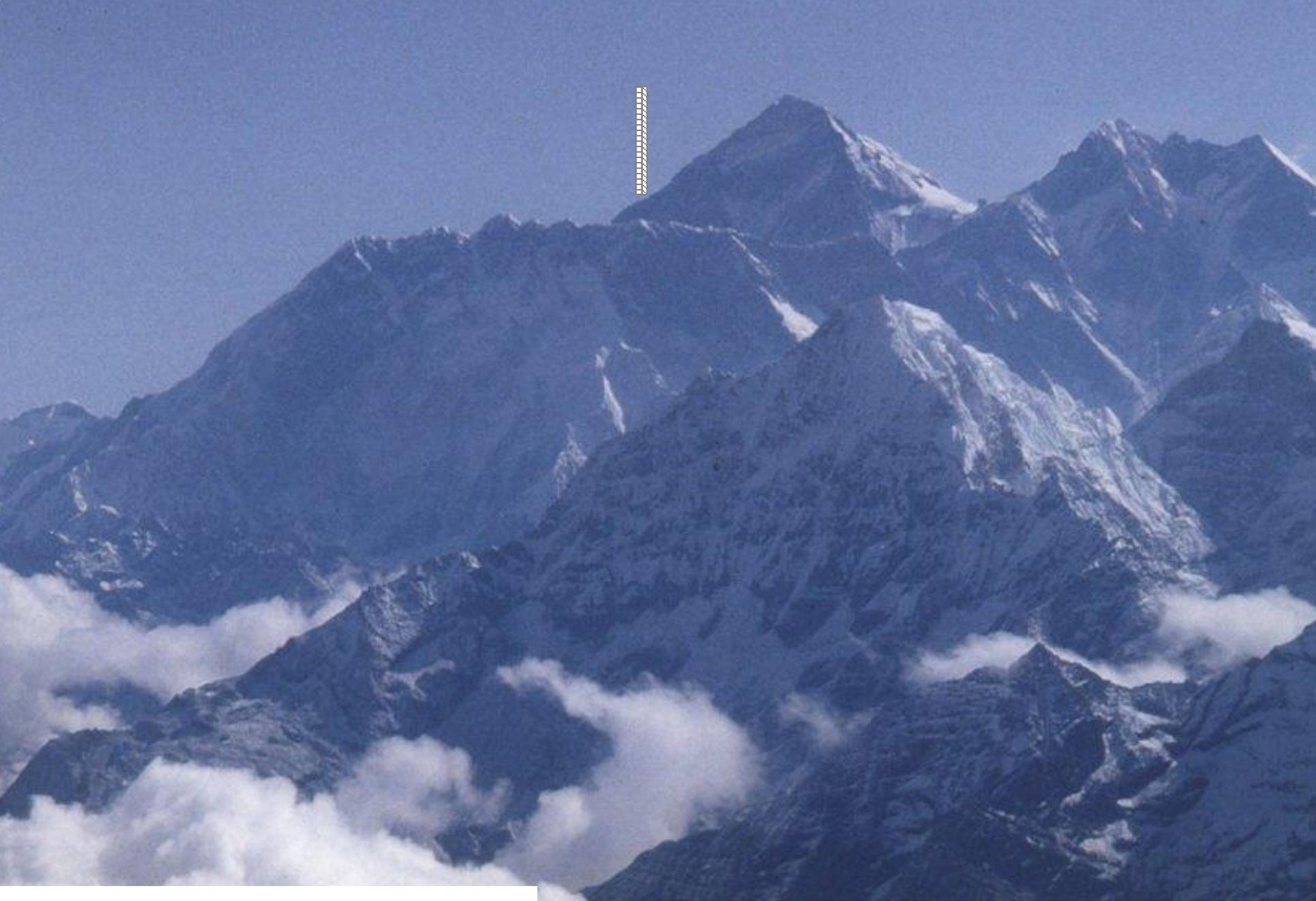


Slide Credit: Nister



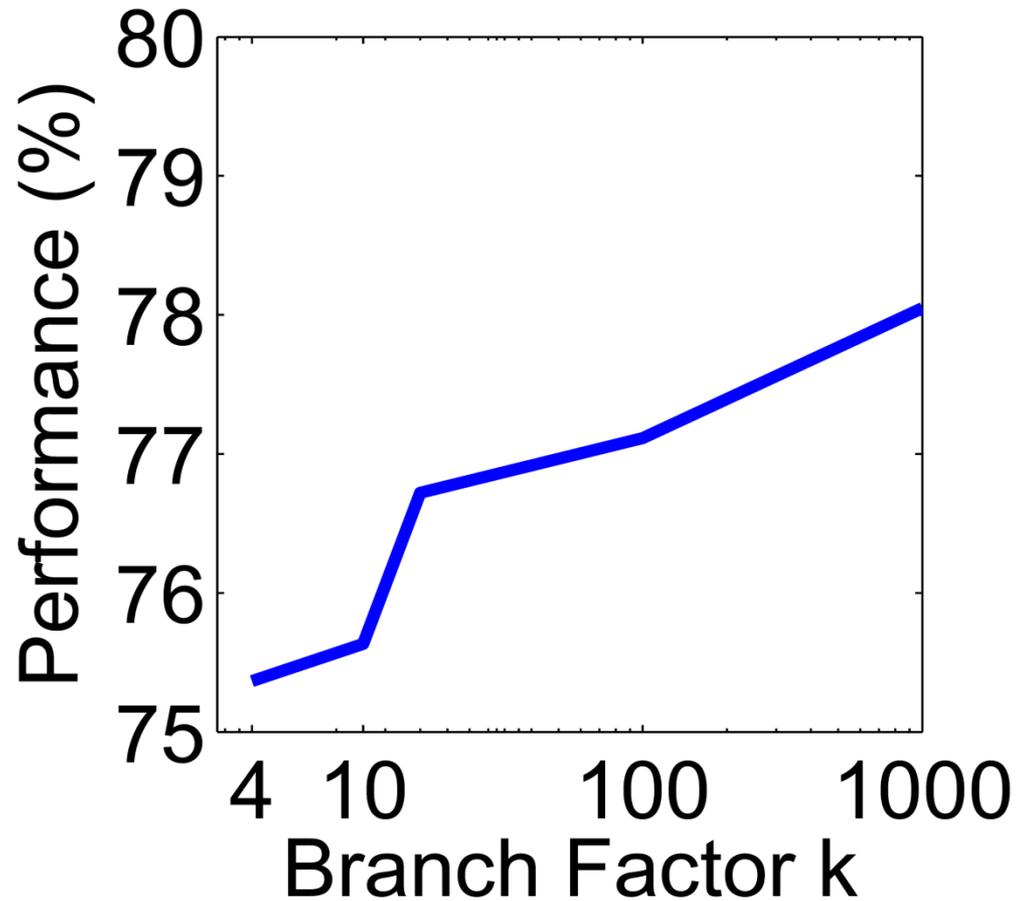
Slide Credit: Nister





Slide Credit: Nister

# Higher branch factor works better (but slower)



# Visual words/bags of words

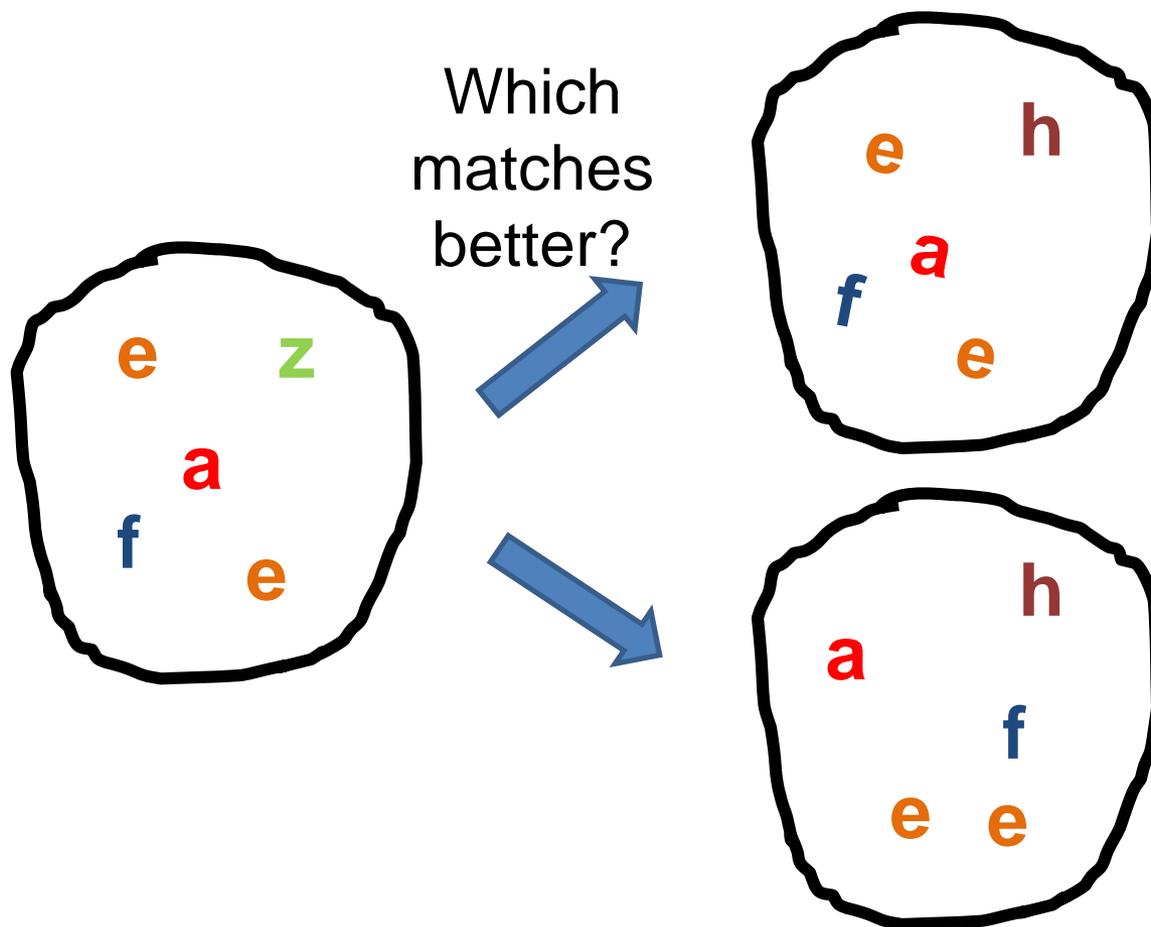
- + flexible to geometry / deformations / viewpoint
- + compact summary of image content
- + provides fixed dimensional vector representation for sets
- + very good results in practice
  
- background and foreground mixed when bag covers whole image
- optimal vocabulary formation remains unclear
- basic model ignores geometry – must verify afterwards, or encode via features

# Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

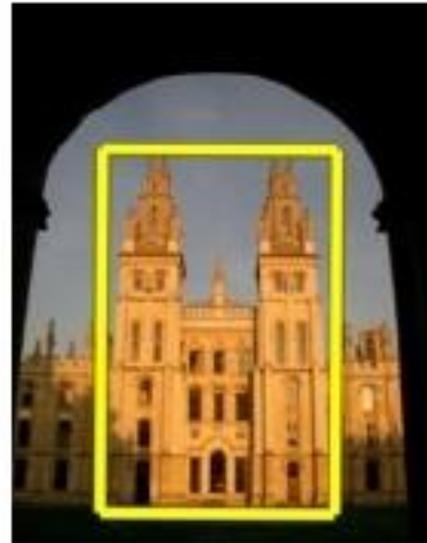
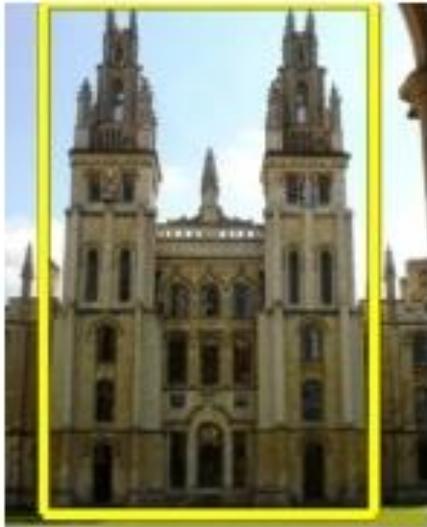
# Can we be more accurate?

So far, we treat each image as containing a “bag of words”, with no spatial information



# Can we be more accurate?

So far, we treat each image as containing a “bag of words”, with no spatial information



Real objects have consistent geometry

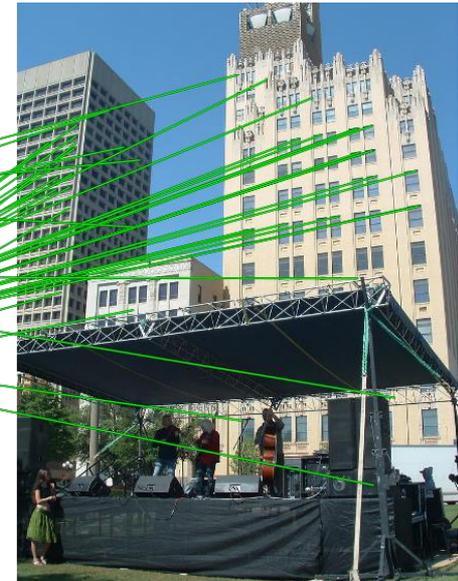
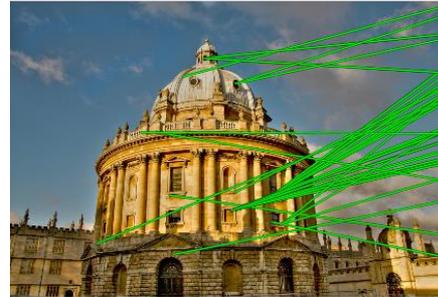
# Spatial Verification

Query



DB image with high BoW  
similarity

Query

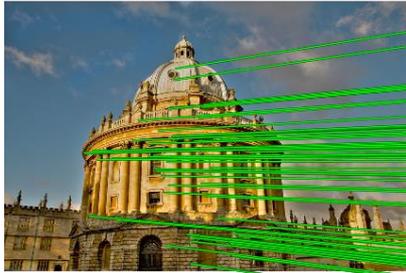


DB image with high BoW  
similarity

Both image pairs have many visual words in common.

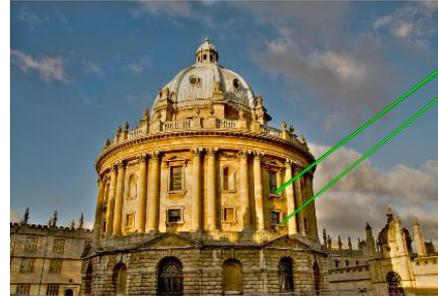
# Spatial Verification

Query



DB image with high BoW  
similarity

Query



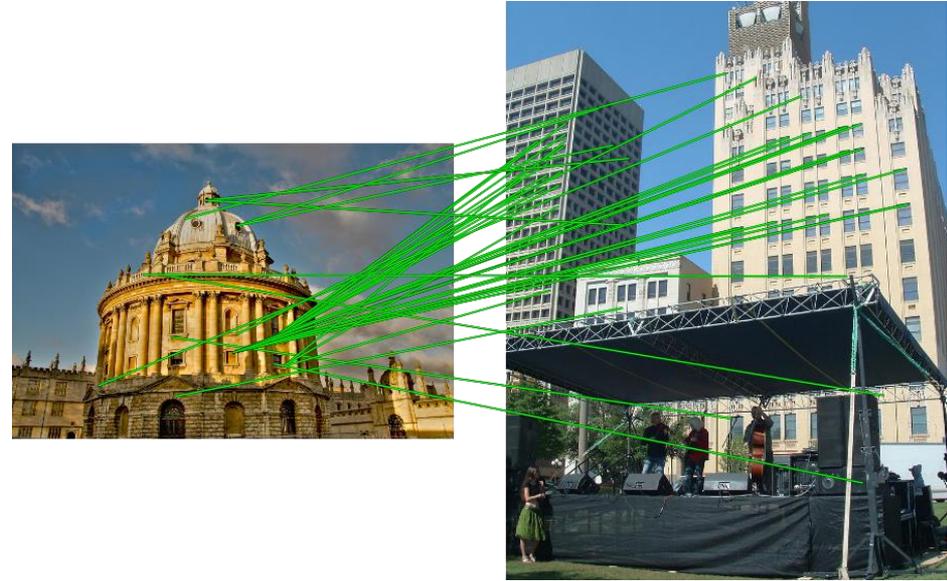
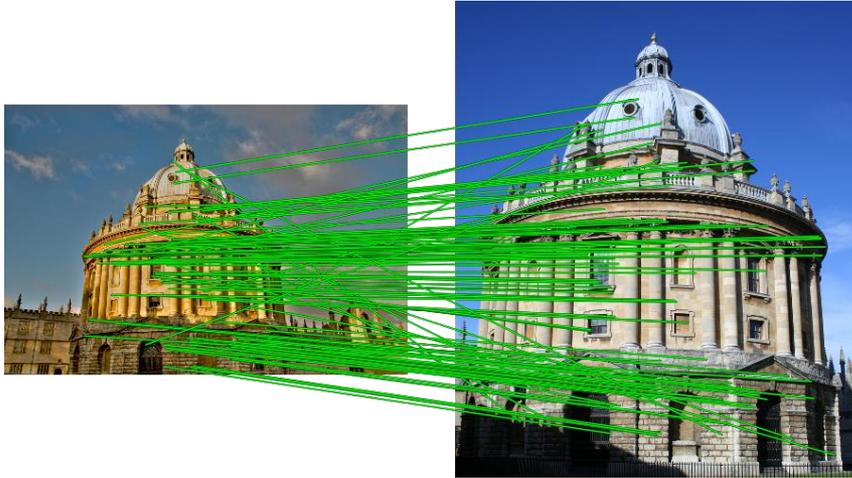
DB image with high BoW  
similarity

Only some of the matches are mutually consistent

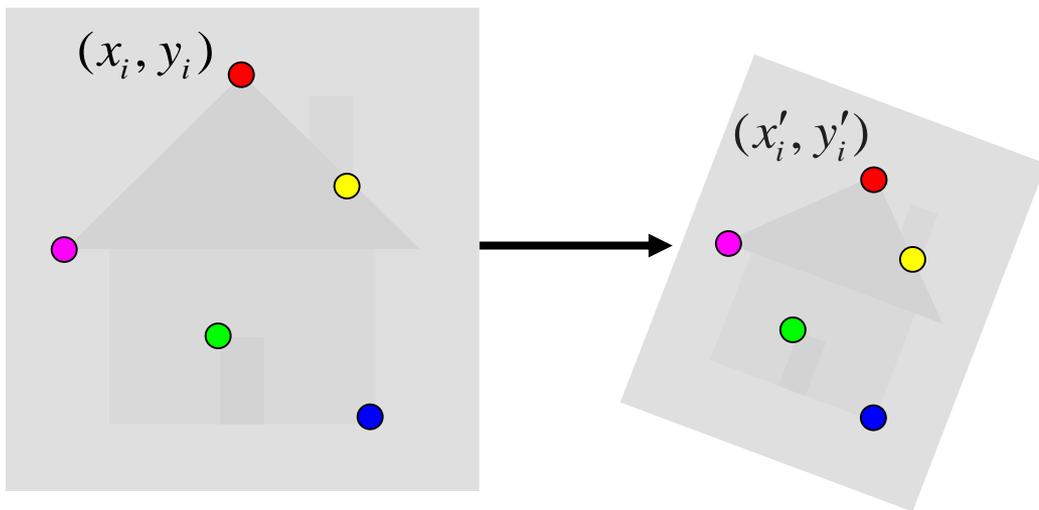
# Spatial Verification: two basic strategies

- RANSAC
  - Typically sort by BoW similarity as initial filter
  - Verify by checking support (inliers) for possible transformations
    - e.g., “success” if find a transformation with  $> N$  inlier correspondences
- Generalized Hough Transform
  - Let each matched feature cast a vote on location, scale, orientation of the model object
  - Verify parameters with enough votes

# RANSAC verification



# Recall: Fitting an affine transformation

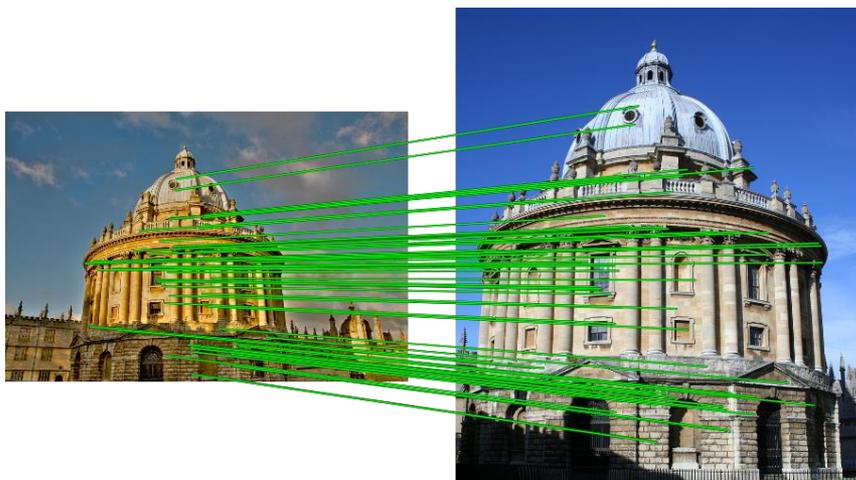
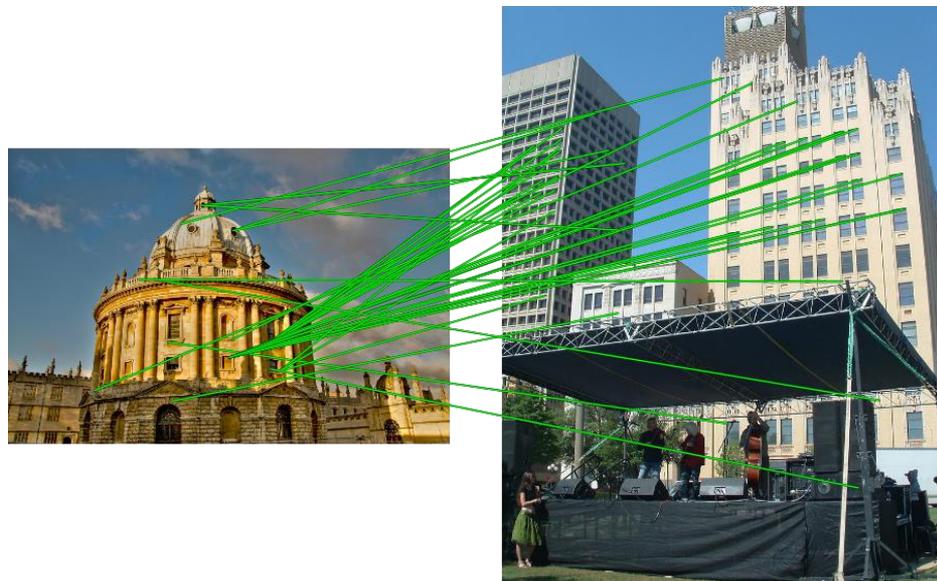
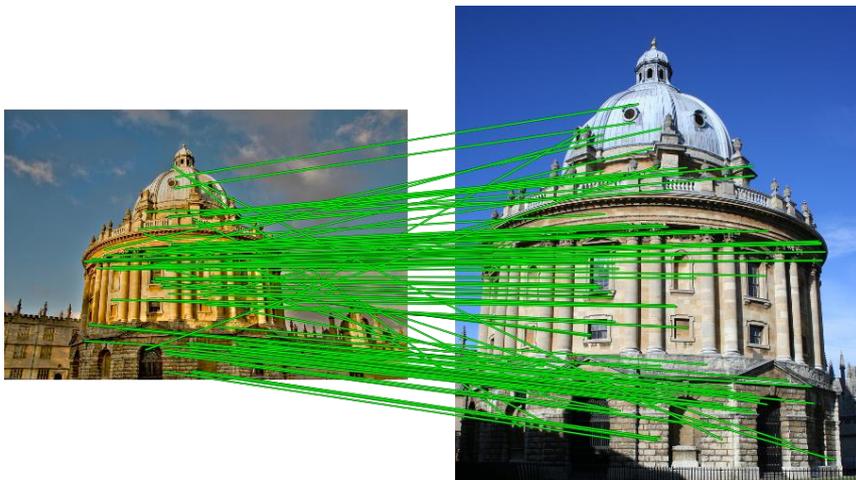


Approximates viewpoint changes for roughly planar objects and roughly orthographic cameras.

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

$$\begin{bmatrix} x_i & y_i & \dots & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} \dots \\ x'_i \\ y'_i \\ \dots \end{bmatrix}$$

# RANSAC verification



# Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

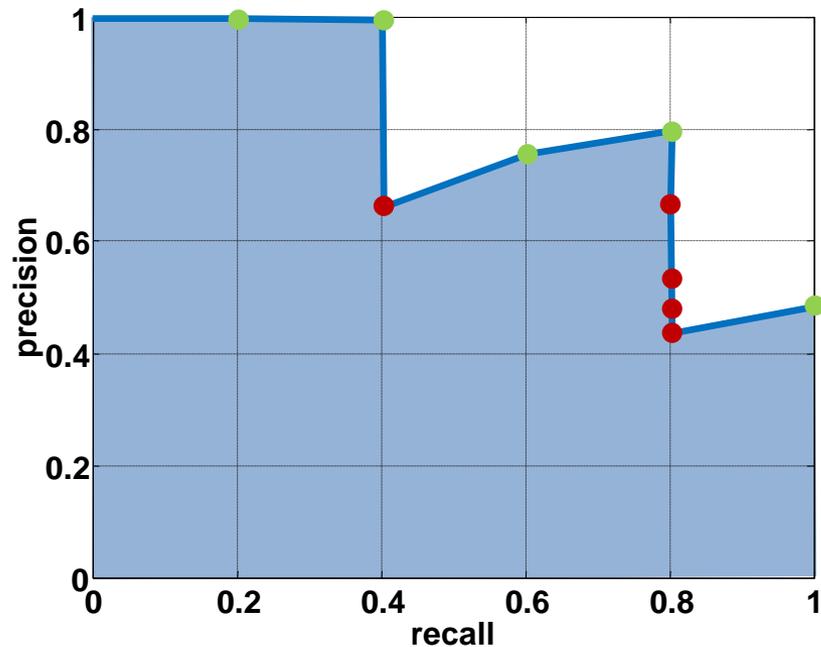
# Scoring retrieval quality



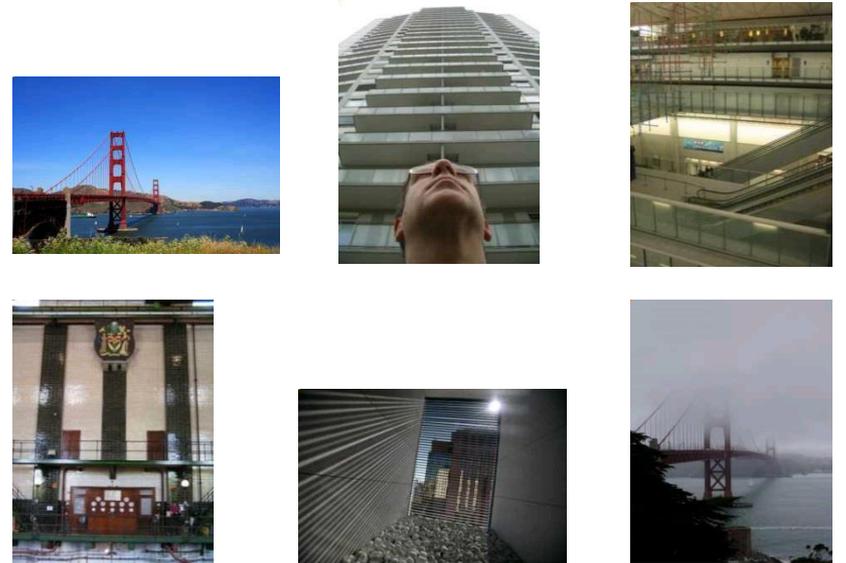
Query

Database size: 10 images  
Relevant (total): 5 images

precision =  $\frac{\text{\#relevant}}{\text{\#returned}}$   
recall =  $\frac{\text{\#relevant}}{\text{\#total relevant}}$



Results (ordered):



# What else can we borrow from text retrieval?

## Index

"Along I-75," From Detroit to Florida; *inside back cover*  
"Drive I-95," From Boston to Florida; *inside back cover*  
1929 Spanish Trail Roadway; 101-102,104  
511 Traffic Information; 83  
A1A (Barrier Is) - I-95 Access; 86  
AAA (and CAA); 83  
AAA National Office; 88  
Abbreviations,  
    Colored 25 mile Maps; cover  
    Exit Services; 196  
    Travelogue; 85  
Africa; 177  
Agricultural Inspection Stns; 126  
Ah-Tah-Thi-Ki Museum; 180  
Air Conditioning, First; 112  
Alabama; 124  
Alachua; 132  
    County; 131  
Alafia River; 143  
Alapaha, Name; 126  
Alfred B MacLay Gardens; 106  
Alligator Alley; 154-155  
Alligator Farm, St Augustine; 169  
Alligator Hole (definition); 157  
Alligator, Buddy; 155  
Alligators; 100,135,138,147,156  
Anastasia Island; 170  
Anhaica; 108-109,146  
Apalachicola River; 112  
Appleton Mus of Art; 136  
Aquifer; 102  
Arabian Nights; 94  
Art Museum, Ringling; 147  
Aruba Beach Cafe; 183  
Aucilla River Project; 106  
Babcock-Web WMA; 151  
Bahia Mar Marina; 184  
Baker County; 99  
Barefoot Mailmen; 182  
Barge Canal; 137  
Bee Line Expy; 80  
Belz Outlet Mall; 89  
Bernard Castro; 136  
Big "I"; 165  
Big Cypress; 155,158  
Big Foot Monster; 105  
Butterfly Center, McGuire; 134  
CAA (see AAA)  
CCC, The; 111,113,115,135,142  
Ca d'Zan; 147  
Caloosahatchee River; 152  
    Name; 150  
Canaveral Natnl Seashore; 173  
Cannon Creek Airpark; 130  
Canopy Road; 106,169  
Cape Canaveral; 174  
Castillo San Marcos; 169  
Cave Diving; 131  
Cayo Costa, Name; 150  
Celebration; 93  
Charlotte County; 149  
Charlotte Harbor; 150  
Chautauqua; 116  
Chipley; 114  
    Name; 115  
Choctawatchee, Name; 115  
Circus Museum, Ringling; 147  
Citrus; 88,97,130,136,140,180  
CityPlace, W Palm Beach; 180  
City Maps,  
    Fl Lauderdale Expwys; 194-195  
    Jacksonville; 163  
    Kissimmee Expwys; 192-193  
    Miami Expressways; 194-195  
    Orlando Expressways; 192-193  
    Pensacola; 26  
    Tallahassee; 191  
    Tampa-St. Petersburg; 63  
    St. Augustine; 191  
Civil War; 100,108,127,138,141  
Clearwater Marine Aquarium; 187  
Collier County; 154  
Collier, Barron; 152  
Colonial Spanish Quarters; 168  
Columbia County; 101,128  
Coquina Building Material; 165  
Corkscrew Swamp, Name; 154  
Cowboys; 95  
Crab Trap II; 144  
Cracker, Florida; 88,95,132  
Crosstown Expy; 11,35,98,143  
Cuban Bread; 184  
Dade Battlefield; 140  
Dade, Maj. Francis; 139-140,161  
Dania Beach Hurricane; 184  
Driving Lanes; 85  
Duval County; 163  
Eau Gallie; 175  
Edison, Thomas; 152  
Eglin AFB; 116-118  
Eight Reale; 176  
Ellenton; 144-145  
Emanuel Point Wreck; 120  
Emergency Callboxes; 83  
Epiphytes; 142,148,157,159  
Escambia Bay; 119  
    Bridge (I-10); 119  
    County; 120  
Estero; 153  
Everglade; 90,95,139-140,154-160  
    Draining of; 156,181  
    Wildlife MA; 160  
    Wonder Gardens; 154  
Falling Waters SP; 115  
Fantasy of Flight; 95  
Fayer Dykes SP; 171  
Fires, Forest; 166  
Fires, Prescribed ; 148  
Fisherman's Village; 151  
Flagler County; 171  
Flagler, Henry; 97,165,167,171  
Florida Aquarium; 186  
Florida,  
    12,000 years ago; 187  
    Cavern SP; 114  
    Map of all Expressways; 2-3  
    Mus of Natural History; 134  
    National Cemetery ; 141  
    Part of Africa; 177  
    Platform; 187  
    Sheriff's Boys Camp; 126  
    Sports Hall of Fame; 130  
    Sun 'n Fun Museum; 97  
    Supreme Court; 107  
Florida's Turnpike (FTP); 178,189  
25 mile Strip Maps; 66  
    Administration; 189  
    Coin System; 190  
    Exit Services; 189  
    HEFT; 76,161,190  
    History; 189  
    Names; 189  
    Service Plazas; 190  
    Spur SR91; 76

China is forecasting a trade surplus of \$90bn (£51bn) to \$100bn this year, a threefold increase on 2004's \$32bn. The Commerce Ministry said the surplus would be created by a predicted 30% increase in exports to \$750bn,

compared with \$566bn. The surplus will annoy the US because China's deliberate policy is to agree to a yuan is also needed to demand so much country. China's yuan against the dollar and permitted it to trade within a narrow band but the US wants the yuan to be allowed to trade freely. However, Beijing has made it clear that it will take its time and tread carefully before allowing the yuan to rise further in value.

**China, trade, surplus, commerce, exports, imports, US, yuan, bank, domestic, foreign, increase, trade, value**

# *tf-idf* weighting

- Term frequency – inverse document frequency
- Describe frame by frequency of each word within it, downweight words that appear often in the database
- (Standard weighting for text retrieval)

Number of occurrences of word  $i$  in document  $d$

Number of words in document  $d$

$$t_i = \frac{n_{id}}{n_d} \log \frac{N}{n_i}$$

Total number of documents in database

Number of documents word  $i$  occurs in, in whole database

# Query expansion

Query: ***golf green***

Results:

- How can the grass on the ***greens*** at a ***golf*** course be so perfect?
- For example, a skilled ***golfer*** expects to reach the ***green*** on a par-four hole in ...
- Manufactures and sells synthetic ***golf*** putting ***greens*** and mats.

Irrelevant result can cause a `topic drift`:

- Volkswagen ***Golf***, 1999, ***Green***, 2000cc, petrol, manual, , hatchback, 94000miles, 2.0 GTi, 2 Registered Keepers, HPI Checked, Air-Conditioning, Front and Rear Parking Sensors, ABS, Alarm, Alloy

# Query Expansion

Results

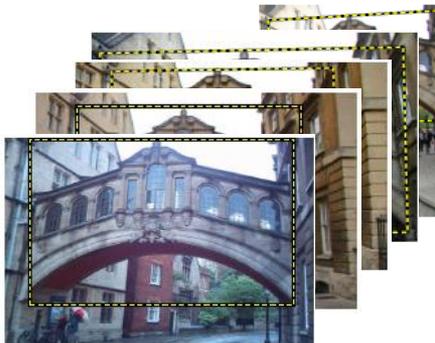


Query image

Spatial verification



New results



New query



Chum, Philbin, Sivic, Isard, Zisserman: Total Recall..., ICCV 2007

Slide credit: Ondrej Chum

# Recognition via alignment

## **Pros:**

- Effective when we are able to find reliable features within clutter
- Great results for matching specific instances

## **Cons:**

- Scaling with number of models
- Spatial verification as post-processing – not seamless, expensive for large-scale problems
- Not suited for category recognition.

# Summary

- **Matching local invariant features**
  - Useful not only to provide matches for multi-view geometry, but also to find objects and scenes.
- **Bag of words** representation: quantize feature space to make discrete set of visual words
  - Summarize image by distribution of words
  - Index individual words
- **Inverted index**: pre-compute index to enable faster search at query time
- **Recognition of instances via alignment**: matching local features followed by spatial verification
  - Robust fitting : RANSAC, GHT

# Lessons from a Decade Later

- For *Category* recognition (project 4)
  - Bag of Feature models remained the state of the art until Deep Learning.
  - Spatial layout either isn't that important or its too difficult to encode.
  - Quantization error is, in fact, the bigger problem. Advanced feature encoding methods address this.
  - Bag of feature models are nearly obsolete. At best they seem to be inspiring tweaks to deep models e.g. NetVLAD.

# Lessons from a Decade Later

- For *instance* retrieval (this lecture)
  - deep learning is taking over.
  - learn better local features (replace SIFT) e.g. MatchNet
  - or learn better image embeddings (replace the histograms of visual features) e.g. Vo and Hays 2016.
  - or learn to do spatial verification e.g. DeTone, Malisiewicz, and Rabinovich 2016.
  - or learn a monolithic deep network to recognition all locations e.g. Google's PlaNet 2016.